

SYSTEMES

CP/M PLUS SUR AMSTRAD CPC, PCW ET COMMODORE 128

YVON DARGERY

ADDITIF

CP/M PLUS SUR AMSTRAD CPC, PCW ET COMMODORE 128

AMSTRAD OU COMMODORE SOUS CP/M PLUS

CP/M Plus est un système d'exploitation conçu par Digital Research pour piloter des ordinateurs fonctionnant sous les microprocesseurs Z 80 ou similaires.

Il ne s'agit donc pas d'un programme développé spécifiquement par un constructeur d'ordinateurs pour sa machine. Quel que soit l'ordinateur qui le fait tourner, CP/M Plus est le même produit.

Il n'existe donc pas de différence fondamentale dans l'utilisation et la syntaxe des ordres de CP/M selon que l'on utilise une machine Commodore ou Amstrad.

Contrairement à d'autres systèmes d'exploitation, CP/M n'est pas fourni par son constructeur avec le programme de formattage des disquettes (les formats de disquettes sont trop différents sous CP/M). De ce fait, il y a des différences d'une machine à l'autre pour formatter, les ordres vous sont donnés dans le mode d'emploi de votre machine : sur Amstrad CPC 6128 ou 8256, vous utiliserez l'ordre DISKIT par exemple.

LES COMMANDES TRANSITOIRES CONSTRUCTEUR

Ces commandes, très utiles sinon indispensables, sont spécifiques d'un ordinateur. Elles sont, en général, liées à l'aspect matériel de la machine comme le formatage d'une disquette ou la copie piste à piste d'une disquette.

Il arrive que certains constructeurs donnent avec leurs machines des commandes transitoires particulièrement performantes, qui pallient la pauvreté de certaines commandes standard (comme les éditeurs de texte par exemple). Certains utilitaires apportent un "plus" très net à la machine en permettant de tirer parti, au maximum, de ses possibilités (comme la création de disques virtuels, par exemple sur les machines 8 bits ayant une taille mémoire de plus de 64 Ko).

LES COMMANDES SPÉCIFIQUES DE L'AMSTRAD

AMSDOS

C'est un ordre compris par CP/M et qui permet de revenir sous le contrôle du système d'exploitation spécifique de l'Amstrad : l'AMSDOS.

L'AMSDOS comporte des ordres de gestion des disquettes mais aussi les ordres du BASIC, c'est donc un système d'exploitation plus complexe que CP/M mais moins sophistiqué dans la gestion spécifique de l'environnement du système physique (périphériques, fichiers...).

DISKIT3

CP/M Plus

Il s'agit d'un utilitaire de formatage et de copie de disquette.

DISKIT fonctionne par dialogue simple à utiliser, n'oubliez pas que dans le système de gestion de l'Amstrad reconnaît un disque B même si vous ne possédez pas de second lecteur, il vous demandera simplement de changer le disque dans le lecteur lorsqu'il aura besoin du second disque.

Dans le cas de l'utilisation d'un seul lecteur de disque, attention à ne pas mélanger les deux disques (source et destination). Cette erreur est irréparable d'où la nécessité de marquer clairement les deux disques.

Dans le cas d'une copie de disquettes sur une disquette vierge, le système lancera d'abord l'opération de formatage.

Le truc

Pour copier une disquette, lancez uniquement l'opération de copie, la combinaison formatage + copie est plus rapide que le formatage sur ordre suivi de la copie sur ordre.

DISKIT2

CP/M 2.2

Utilitaire de fonctionnement identique à DISKIT3 mais spécifique au format du CP/M 2.2.

LANGUAGE

Cet ordre permet de reconfigurer le système en fonction de spécificités de certaines langues (accents, signes financiers...).

Il est à utiliser avec prudence et de toutes façons, une machine achetée en France a été configurée avec les caractères nationaux ; cependant, il peut être indispensable de faire des modifications grâce à cet ordre, dans ce cas, il faut penser à configurer en conséquence l'imprimante.

FILECOPY

CP/M 2 uniquement

Cet utilitaire vous permet de copier des fichiers d'une disquette sur une autre si vous ne disposez que d'un seul lecteur.

Le programme vous demande de changer les disquettes si nécessaire.

Cet utilitaire est inutile avec les nouvelles configurations qui avec l'ordre PIP référencié à deux noms de lecteurs fera exactement le même travail de manière bien plus simple.

PALETTE

Si vous disposez, avec votre Amstrad, d'un moniteur couleur, vous pouvez sous CP/M modifier les couleurs du texte et du fond.

Par défaut, le texte est en blanc sur fond bleu.

Les couleurs sont définies par l'ordre PALETTE suivi de deux nombres de 0 à 63, le premier définit le fond, le second définit le texte.

Par exemple, PALETTE 32,8 débute une période d'écriture en rouge (8) sur fond vert (32).

SETKEYS

Vous avez la possibilité de redéfinir le contenu des touches si vous écrivez dans un fichier texte et pour chaque ligne le code de la touche suivi du contenu de ce qu'il doit générer placé entre parenthèses.

La redéfinition des touches se fait par l'ordre

SETKEYS fich

où fich est le nom du fichier contenant les codes des touches.

Les fichiers livrés avec votre système sont KEYS.CCP pour la configuration standard CP/M, KEYS.DRL pour la configuration LOGO, KEYS.WP pour la configuration de certains traitements de texte.

SETLST

Vous avez la possibilité, sous CP/M, d'envoyer une chaîne d'initialisation à l'imprimante ; la chaîne est contenue dans un fichier texte, elle est envoyée par l'ordre

SETLST fich

où fich est le nom du fichier contenant la chaîne.

Par exemple, l'imprimante, écrivant en caractères élargis avec l'envoi du code décimal 14, écrira dans ce format à partir du moment où l'ordre SETLST LARGE aura été lancé, si le fichier "LARGE" contient la chaîne ^' 14' (^ pour control, ou l'ESC de l'imprimante).

SETSIO

Si votre Amstrad est doté de l'interface RS 232, vous pouvez la reprogrammer avec l'ordre SETSIO (en plus de l'ordre DEVICE) sous la forme

SETSIO (qui donne la configuration).

SETSIO, RX 2400, TX 300, PARITY NONE, STOP 1, BITS 8,
HANDSHAKE ON, XOFF OFF (qui donne une nouvelle configuration).

SETSIO, XOFF ON (qui modifie un des éléments).

CP/M PLUS SUR COMMODORE 128

Connaissez-vous toute la collection Commodore 128 chez P.S.I. ?

Système

— Clefs pour Commodore 128 : 1 - Mode 64 — Daniel-Jean David

A paraître

Programmation BASIC :

— 102 programmes pour Commodore 128 — Jacques Deconchat

Maîtrise du BASIC :

— Commodore 128 : Méthodes pratiques — Jacques Boisgontier
— Graphiques et sons du Commodore 128 — Daniel-Jean David

Système :

— Clefs pour Commodore 128 : 2 - Mode 128 — Jean-François Sehan

Pour tout problème rencontré dans les ouvrages P.S.I.
vous pouvez nous contacter au numéro ci-dessous :

Numéro Vert/Appel Gratuit en France

05 21 22 01

(Composer tous les chiffres, même en région parisienne)

La loi du 11 mars 1957 n'autorisant, aux termes des alinéas 2 et 3 de l'article 41, d'une part, que les « copies ou reproductions strictement réservées à l'usage privé du copiste et non destinées à une utilisation collective », et, d'autre part, que les analyses et les courtes citations dans un but d'exemple et d'illustration, « toute représentation ou reproduction intégrale, ou partielle, faite sans le consentement de l'auteur ou de ses ayants droit ou ayants cause, est illicite » (alinéa 1er de l'article 40).

Cette représentation ou reproduction, par quelque procédé que ce soit, constituerait donc une contrefaçon sanctionnée par les articles 425 et suivants du Code Pénal.

© *Editions du P.S.I., B.P. 86, 77402 Lagny Cedex, 1986*
ISBN : 2-86595-327-0

SYSTÈMES D'EXPLOITATION

CP/M PLUS SUR COMMODORE 128

YVON DARGERY



ÉDITIONS DU P.S.I.
1986

Yvon **Dargery** est un ancien enseignant. Attaché à l'Institut national d'education populaire du ministère de la Jeunesse et des Sports, il est spécialisé dans la pédagogie pour les disciplines scientifiques. Par ailleurs, conseiller en informatique auprès de grandes entreprises, il travaille à la mise en place cohérente d'applications professionnelles autour d'ordinateurs personnels.

SOMMAIRE

AVANT-PROPOS

LE DOS

CP/M est un DOS, c'est-à-dire un Disk Operating System, système de gestion de disquettes.

Il a pour mission de soulager l'utilisateur des opérations fastidieuses, mais nécessaires, pour utiliser au mieux l'espace disponible sur la disquette afin d'y stocker des informations.

Il doit, en outre, assurer l'harmonie de fonctionnement entre l'ordinateur proprement dit et ses périphériques (écran, clavier, mémoire...) ; c'est pour cela que l'on parle de "système d'exploitation", terme plus général que DOS, et qui reflète mieux la réalité.

C'est le DOS qui tiendra à jour la liste des fichiers avec leur taille et la place physique qu'ils occupent et c'est également à lui que le microprocesseur déléguera la tâche d'écrire ou de lire sur le disque.

En cas d'insuffisance de place, c'est toujours lui qui découpera un fichier en morceaux pour utiliser les petits espaces disponibles et la taille de la disquette au mieux.

Bref, un serviteur fidèle au service de l'utilisateur.

CP/M

Le système d'exploitation est constitué en fait de plusieurs parties, en ce qui concerne la position dans la mémoire mais aussi en termes de fonctions.

Voici ces zones classées dans l'ordre de leur apparition sur la carte mémoire :

– **la zone du système** (ou *PAGE Ø*) : elle n'appartient pas exactement à CP/M, elle caractérise l'ordinateur hôte. Elle contient notamment les utilitaires de départ avant le chargement de CP/M ;

– **la zone TPA** (pour *Transian Program Area, zone de programmes transitoire*) : c'est la zone laissée accessible à l'utilisateur. Ici seront écrits les programmes, les utilitaires, les données qui n'appartiennent pas explicitement à CP/M ;

– **la zone CCP** (pour *Console Command Processor, contrôle des commandes de l'opérateur*) : elle est chargée de gérer les commandes qui sont envoyées par l'opérateur au moyen de la console (écran-clavier, en l'occurrence du clavier) ; les commandes reconnues par CCP sont les "commandes résidentes" de CP/M que nous verrons plus loin.

Cette zone est immédiatement placée après la zone utilisateur, il peut donc arriver qu'elle soit détruite si la place est insuffisante. Dans ce cas, les ordres CP/M ne seront pas reconnus lors du retour au système et un démarrage à chaud sera nécessaire ;

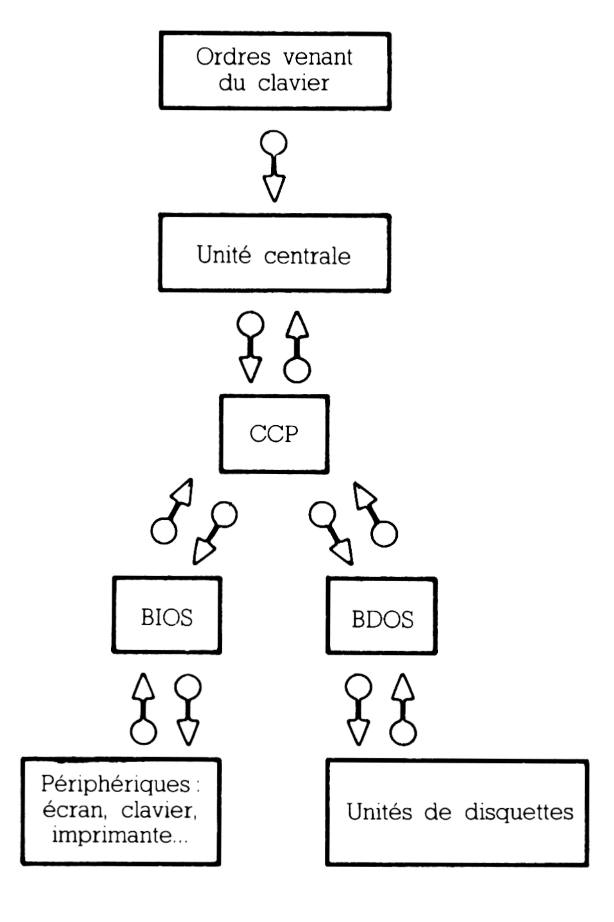
– **la zone BDOS** (pour *Basic Disk Operating System*) : elle contient les programmes propres de gestion de la lecture et de l'écriture des fichiers sur disque ; elle est indispensable au système et ne peut donc pas être détruite même en cas de manque de place en mémoire centrale ;

– **la zone BIOS** (pour *Basic Input Output System*) : C'est celle qui permet à tel ou tel système ou à telle ou telle configuration de tourner sous CP/M ; elle est adaptée par le constructeur en fonction des caractéristiques propres de sa machine (type d'écran, de clavier, de périphériques spécifiques...).

C'est cette zone que vous devrez modifier si vous achetez un CP/M standard et que vous désirez le faire tourner... Je vous souhaite alors bien du plaisir si vous connaissez mal les caractéristiques des périphériques que vous aurez à utiliser ou si vous n'êtes pas né avec un couple d'ordinateurs comme parents.

Les deux dernières zones (BDOS et BIOS) sont appelées FDOS (*Functional Disk Operating System*), c'est-à-dire l'ensemble des ordres de gestion du système.

Les trois dernières zones (CCP, BDOS, BIOS) forment le CP/M proprement dit.



LA FAMILLE CP/M

CP/M est le nom d'une grande famille de systèmes d'exploitation de disquettes et signifie (traduit en français) contrôle de processus pour microprocesseurs.

CP/M est né en 1974. A cette époque, Gary Kildall, employé par Intel (importante société fabriquant des circuits intégrés), travaillait à la mise au point d'un utilitaire aidant à gérer le fonctionnement des disquettes des ordinateurs qu'il utilisait.

Le fonctionnement tout à fait satisfaisant du CP/M, sa place unique sur le marché, mais surtout la progression du nombre des micro-ordinateurs firent rapidement son succès.

Pour commercialiser son produit, Kildall fonda une société : Digital Research, chargée de la commercialisation et de la recherche opérationnelle autour du produit.

LES VERSIONS DE CP/M

CP/M, d'abord créé pour fonctionner sur le microprocesseur 8080 d'Intel, puis sur le Z80, processeur 8 bits de grande diffusion, a subi de nombreux remaniements qui se concrétisent par des numéros de versions, voire des noms différents.

☐ **CP/M 80** : version 8 bits du DOS, a de nombreuses versions notées par un nombre décimal :
– la version 1.3 est la première version commercialisée en 1985, elle a pratiquement disparu ;
– la version 2 et 2.2 est la version actuelle de nombreux micro-ordinateurs de gestion tournant sous des microprocesseurs de 8 bits ; ils sont de plus en plus rares ;
– la version 3.0, communément appelée **CP/M Plus**, est la dernière version de CP/M pour microprocesseur Z80 ; c'est la version qui équipe les machines semi-professionnelles de dernière génération (Amstrad, Commodore, etc.).

☐ **MP/M** : version multi-utilisateur de CP/M, c'est-à-dire qu'un ordinateur central fonctionne avec MP/M, cependant que des consoles (écran/clavier) peuvent travailler en même temps, MP/M se chargeant de gérer l'accès des utilisateurs, les uns après les autres.

☐ **CP/M 86** : version 16 bits, donc plus puissante de CP/M. Elle a eu moins de succès que CP/M 80, peut-être parce que moins moderne que ses concurrentes ; elle est disponible sur option sur les IBM-PC et compatibles. Cette version de CP/M a été écrite pour fonctionner avec des microprocesseurs héritiers du 8080, c'est-à-dire les 8088 ou 8086 qui équipent la quasi-totalité des micro-ordinateurs modernes de petite gestion.

☐ **Concurrent CP/M (ou CCP/M)** : version réseau de CP/M 86, créée à l'origine pour les réseaux locaux d'entreprise, elle devait permettre de faire travailler en même temps plusieurs micro-ordinateurs, chacun accédant aux ressources des autres imprimantes, disquettes, disque durs...).

D'une version à l'autre sont apparus des changements, souvent mineurs pour l'utilisateur, de nouvelles fonctionnalités ; surtout l'on a vu apparaître les résultats d'une recherche de meilleure fiabilité et des possibilités accrues.

UNE AIDE PRÉCIEUSE

CP/M Plus (ainsi que CP/M 86) offre une aide en ligne. Cette aide est obtenue par la frappe de l'ordre HELP (ou AIDE dans le cas de certains constructeurs français).

Le système passe en mode HELP ; il donne alors en début de ligne le signe :

HELP

indiquant qu'il attend la frappe du mot sur lequel on désire des explications.

L'écran affiche alors un texte explicatif (le plus souvent en anglais) sur le mot spécifié.

L'aide est arborescente ; on peut donc demander l'explication d'une option d'un ordre.

Par exemple,

HELP PIP OPTIONS

donnera la liste commentée des options de l'ordre PIP.

Pour sortir du mode HELP, il suffit de frapper la touche **RETURN** sans argument au symbole HELP>.

LES FICHIERS DE CP/M

La partie DOS de CP/M est donc chargée de gérer l'espace disponible dans les disquettes. L'organisation des informations s'opère de manière semblable à la gestion d'une bibliothèque. Le système a des données contenues dans des fichiers comparables aux volumes d'une bibliothèque ; les références (nom, place, taille...) sont placées dans un directory ou catalogue, comme dans le catalogue d'une bibliothèque. Comme pour les livres, toute modification d'un fichier est consignée dans le directory.

Un fichier est donc un terme générique pour le système ; il désigne l'ensemble cohérent d'informations regroupées sous un seul nom. Prenons l'exemple d'un programme de traitement de texte avec lequel on écrit un rapport ; le programme s'appelle par exemple TEXTUS, le rapport RAPP1. Pour le DOS, il y a deux fichiers, TEXTUS et RAPP1, et il n'y a pas, au moins dans la gestion de l'espace, de différence entre les deux.

LE NOM DES FICHIERS

LE NOM

Tableau 1

Caractères à éviter ou interdits dans les noms de fichiers:

<	>	$	,	!	[	]	Tabulateur	:	.	
;	*	?	+	-	(	)	/	=		

Pour CP/M, comme pour la plupart des DOS, le nom d'un fichier est constitué de trois parties :

– le nom du lecteur où il se trouve : *Exemple B :* signifie lecteur de disquette B. Le nom du disque est superflu si le fichier se trouve sur le disque actif ;

– le nom du fichier proprement dit. Il s'agit d'un mot de huit lettres ou chiffres au maximum ; certains autres signes sont autorisés ;

– l'extension du fichier est séparée par un point du nom. Elle aura trois signes au maximum, lettres ou chiffres. Certaines extensions sont réservées au système ; il les met automatiquement.

```
Tableau 2

Noms d'extensions réservées au système:

 .BAS       Programme écrit en BASIC
 .COM       Programme  directement exécutable à l'appel de son  nom
                 seul, en général écrit en assembleur.
 .ASM       Programme source écrit en assembleur
 .SUB       Fichier  contenant une liste d'instructions qui seront
                 exécutées automatiquement avec l'ordre SUBMIT
 .$$$       Fichier  temporaire  créé  par  le  système,  ne  doit
                 théoriquement pas subsister sur le disque, sauf en
                 cas d'incident
 .BAK       Double d'un fichier créé automatiquement par  certains
                 traitements de texte (initiale de BAcKup =  sauve-
                 garde)

D'autre  extensions  sont  spécifiques  à une  machine  ou  à  un
utilitaire  particulier,  consultez votre documentation pour  les
retrouver.
```

Le mot de passe éventuel du fichier est séparé de l'extension par le signe ; . Il permet de faire les opérations sur des fichiers déclarés protégés. Si le mot de passe n'est pas spécifié et le fichier protégé, le système demandera le mot de passe avant de continuer. Si le fichier protégé est un fichier de commande (extension .COM), le mot de passe est indispensable pour l'exécuter.

Par exemple :

- A:PROGR.TXT est le programme de nom PROGR, d'extension TXT sur le disque A (si le lecteur A est actif) ;
- B:PROGRAM.TXT;PASS est le programme sur le lecteur B, de nom PROGRAM, d'extension TXT et de mot de passe PASS ;
- A:PROG.AB.TXT est invalide (deux fois un point) ;
- A:PROG-AB.TXT est invalide (signe - interdit) ;
- B:MONPROGRAMME.TXT;PASS est valide, mais le nom trop long sera tronqué par le système aux huit premiers caractères ; il sera archivé B:MONPROG.TXT;PASS.

LES CARACTÈRES AMBIGUS DANS LE NOM DES FICHIERS (OU CARACTÈRES JOKER OU WILD CARDS)

Il est possible de donner des noms de fichiers comportant des ambiguïtés, notamment pour faire porter un ordre sur plusieurs fichiers.

Le signe ? remplace une lettre ; par exemple, le nom de fichier FICH?AB sera valide pour FICH1AB, FICH9AB, FICHAAB, mais pas pour FICH1.

Deux signes ?? donnent l'ambiguïté pour deux lettres.
Il est possible de combiner le signe ? pour le nom et l'extension : PROG??AB.00? sera valable pour PROGWSAB.001 par exemple, mais pas pour PROGWS.001 ni pour PROGWSAB.010.

Le signe * remplace la chaîne de lettres qui le suit ; par exemple, PROG* est valide pour PROG, pour PROGRAM, pour PROG001 ou tout mot commençant par PROG, quelle que soit sa longueur (dans la limite de la longueur autorisée, c'est-à-dire 8 signes pour le nom, 3 signes pour l'extension).
. sera donc valide pour tous les noms de fichiers.
PROGRAM.* sera valide pour tout fichier de nom PROGRAM, quelle que soit son extension.
PROG*.BAS sera valide pour tout fichier dont le nom commence par PROG, d'extension BAS, donc PROG001.BAS, PROGRAM.BAS, PROG.BAS... mais pas PROG.TXT ni PROTO.BAS.

LES ATTRIBUTS DES FICHIERS

Les noms des fichiers désignent donc une unité d'information pour le système d'exploitation. Ces fichiers sont, en outre, affectés d'attributs qui modifient la façon dont ils seront gérés par le système.

Les attributs de protection

Une protection peut être mise aux fichiers. Selon la hiérarchie de cette protection, les fichiers seront ineffaçables (attribut RO = Read Only), effaçables (RW = Read or Write) ou protégés par un mot de passe (Password : voir l'ordre SET).

Les attributs de directory

Les fichiers peuvent être rendus opaques à l'ordre DIR si on les affecte de l'attribut SYS. Dans ce cas, ils n'apparaissent pas dans la liste correspondant à la commande DIR ; on dit que les fichiers sont Système ; ils sont alors utilisables de n'importe quelle zone USER. L'attribut DIR est l'attribut par défaut, les fichiers sont apparents au directory et appartiennent à une seule zone USER.

LE NOM DES LECTEURS DE DISQUETTES

Le lecteur actif

Les lecteurs sont nommés par une lettre (A, B, C...) ; le lecteur actif est celui sur lequel sont lues les informations, sauf indication contraire.
Le nom du lecteur actif est celui qui apparaît en début de ligne de commande CP/M.

Exemple : A>_ signifie que CP/M attend des ordres et que le lecteur A est actif.
Pour rendre un lecteur actif, il suffit de taper son nom suivi de deux points (:).

Exemple : on frappe B: suivi de la touche **RETURN** ; le message devient B>_, sous réserve, bien sûr, que l'on dispose de deux lecteurs de disquettes.

LA SYNTAXE DES ORDRES

Les ordres de CP/M Plus sont des mots clés, reconnus par le système, et qui déclenchent une action ou une suite d'actions.

Les ordres sont compris du système *en majuscules* seulement. Par sécurité, les lettres frappées en minuscules seront automatiquement traduites en majuscules avant d'être exécutées. Cette particularité est à retenir, notamment dans le cas d'ordre portant sur des fichiers créés sous un autre outil qui accepte les minuscules ; (*ex.* – un fichier sauvé sous BASIC en minuscules ne sera plus exploitable sous CP/M).

LA SYNTAXE GÉNÉRALE DES ORDRES

La syntaxe générale des ordres est la suivante :

ORDRE ARGUMENT [OPTION]

L'ordre, l'argument et l'option sont séparés par un espace.

L'argument peut comprendre plusieurs éléments, parfois séparés par un signe (*ex.* _PIP FICHIER1=FICHIER2 qui recopie le fichier2 sous le nom fichier1).

L'option peut aussi comprendre plusieurs éléments ; dans CP/M Plus, l'option est, en général, placée entre crochets [et].

Argument et option peuvent être *facultatifs :* dans ce cas, ce sont les arguments et les options par défaut qui seront pris (*ex.* DIR donne le directory du disque actif en version condensée).

DES ABRÉVIATIONS

Certains ordres ont des abréviations ; nous en avons vu certaines. En outre, de très nombreux arguments ont aussi des abréviations qu'il est intéressant de connaître, surtout pour des arguments longs (PASSWORD peut s'abréger en PA !).

Expérimentalement, vous découvrirez ces abréviations ; certaines vous sont données dans les Trucs des ordres.

ORDRE NON RECONNU

Si le système reçoit un ordre qu'il ne comprend pas, soit parce que l'orthographe de l'ordre est incorrecte, soit parce que le fichier exécutable n'est pas sur le disque actif, CP/M répond en répétant l'ordre, suivi d'un point d'interrogation ; par exemple, tapez DAR au lieu de DIR : le système répond DAR? et vous rend la main.

LES CONSTITUANTS DE CP/M

CP/M est constitué d'un certain nombre de fichiers. Certains forment le système lui-même : ce sont ceux qui se placent en mémoire à la mise en activité du système d'exploitation. D'autres constituent des commandes externes ; il s'agit de petits programmes utilitaires qui permettent d'effectuer des actions plus complexes que celles possibles sous le système seul.

LE SYSTÈME EST CONSTITUÉ DE TROIS ENSEMBLES

Le **CCP** (pour *Console Command Processor*) est le programme qui règle les échanges entre l'ordinateur (l'unité centrale) et ses périphériques. C'est lui, notamment, qui prendra en compte les touches que vous frapperez au clavier pour les coder et les transmettre à l'unité centrale. Le CCP contient les utilitaires nécessaires pour reconnaître et exécuter les commandes résidentes que nous verrons plus loin.

Le **BIOS** (pour *Basic Input/Output System*) est le programme qui gère les entrées/sorties de base, c'est-à-dire les fonctions qui permettent à CP/M, programme général, indépendant de toute machine, de reconnaître la machine précise que vous utilisez : les codes d'écran, de clavier, etc. Il s'agit donc d'un programme différent d'une machine à l'autre, un peu comme un interprète entre CP/M, sorte de langage universel, et votre machine, avec toutes ses spécificités.

Le **BDOS** (pour *Basic Disk Operating System*) contient quant à lui les programmes de base de gestion des disquettes : le contenu de la disquette active, comment classer ses fichiers, les créer, y lire ou écrire des informations... c'est une des zones prioritaires du système lorsque CP/M est implanté en mémoire.

Ces trois ensembles constituent le CP/M proprement dit, lorsque vous le chargez dans votre ordinateur.

LES COMMANDES DE CP/M

CP/M comporte trois types de commandes qu'il est susceptible d'exécuter : il s'agit des commandes résidentes, des commandes transitoires système et des commandes transitoires constructeur.

LES COMMANDES RÉSIDENTES

Les Anglo-Saxons parlent de BUILT IN commandes, donc commandes résidentes dans le DOS lui-même.

Il s'agit de commandes reconnues par le CCP ; elles sont donc exécutables directement par le système :

- DIR : donne la liste des fichiers ;
- DIRSYS : donne la liste des fichiers système (classés ainsi par l'utilisateur, *a priori*, fichiers sensibles) ;
- ERASE : efface un ou plusieurs fichiers ;
- RENAME : change le nom d'un fichier ;
- TYPE : liste le contenu d'un fichier ;
- USER : permet de changer la zone utilisateur de la disquette.

Les abréviations usuelles sont les suivantes :

DYRSYS = DIRS
ERASE = ERA
RENAME = REN
TYPE = TYP
USER = USE

LES COMMANDES TRANSITOIRES SYSTÈME OU COMMANDES NON RÉSIDENTES

Il s'agit en fait de fichiers contenus sur la disquette et qui contiennent des programmes qui réalisent une ou plusieurs tâches utiles ou indispensables pour l'utilisateur de CP/M. Ces fichiers doivent avoir l'extension .COM pour être directement exécutables.

Ces commandes sont communes à la presque totalité des machines fonctionnant sous CP/M Plus. Elles sont l'œuvre de Digital Research : on peut donc considérer qu'elles font partie du système CP/M.

☐ **DATE** : permet de mettre à jour ou de restituer la date et l'heure connues du système ;

☐ **DEVICE** : permet de visualiser ou de changer les paramètres d'affectation des entrées/sorties (type d'imprimante, vitesse et type de transfert série...) ;

☐ **DIR** : extension à l'ordre DIR résident ; permet de donner des paramètres de tri, de type d'information, etc. ;

☐ **DUMP** : permet de lister un fichier sous forme hexadécimale ;

☐ **ED** : éditeur ligne par ligne ; permet d'intervenir à l'intérieur d'un fichier texte sur disque ;

☐ **ERASE** : extension de l'ordre ERA pour sécuriré ; demande confirmation pour l'effacement ;

☐ **GET** : (et son contraire PUT) ; modifie provisoirement le périphérique de sortie ou d'entrée (de la console vers un fichier, par exemple, pour garder la trace d'une manœuvre) ;

☐ **HELP** : aide arborescente sur les ordres et les commandes de CP/M ;

☐ **INITDIR** : initialisation du directory d'une disquette pour y écrire la date et l'heure de manipulation des fichiers qu'elle contient ;

☐ **PATCH** : permet de modifier certains paramètres de programmes du CP/M grâce à des options chiffrées ;

☐ **PIP** : programme de gestion d'échanges entre périphériques (disquettes, imprimante) ;

☐ **RENAME** : extension de l'ordre résident REN ; permet de confirmer les changements ou de faire des changements de nom avec caractères ambigus ;

☐ **SAVE** : sauvegarde le contenu de la mémoire dans un fichier ;

☐ **SET** : puissant utilitaire de modification des paramètres de fichiers (protection, mots de passe) ;

☐ **SHOW** : donne l'état d'occupation d'une disquette ;

☐ **STAT** : permet de connaître l'encombrement d'un disque et la taille des fichiers, ainsi que de modifier certains paramètres des fichiers. Ancien utilitaire de CP/M 2.2, il a perdu beaucoup d'intérêt avec les nouvelles fonctionnalités de SHOW, de DIR et de SET. Il n'existe pas dans les versions de CP/M Plus ;

☐ **SUBMIT** : utilitaire permettant d'enchaîner les ordres CP/M contenus dans un fichier d'extension .SUB ;

☐ **SID** : outil de mise au point de programmes machine ; c'est un utilitaire délicat à utiliser, puisqu'il travaille directement sur le contenu de la disquette.

LES COMMANDES TRANSITOIRES CONSTRUCTEUR

Ces commandes, très utiles sinon indispensables, sont spécifiques d'un ordinateur. Elles sont, en général, liées à l'aspect matériel de la machine comme le formatage d'une disquette ou la copie piste à piste d'une disquette.

Il arrive que certains constructeurs donnent avec leurs machines des commandes transitoires particulièrement performantes, qui pallient la pauvreté de certaines commandes standard (comme les éditeurs de texte par exemple). Certains utilitaires apportent un "plus" très net à la machine en permettant de tirer parti, au maximum, de ses possibilités (comme la création de disques virtuels, par exemple sur les machines 8 bits ayant une taille mémoire de plus de 64 Ko).

LES TOUCHES DE COMMANDES

Certaines combinaisons de touches ont une action particulière sous CP/M Plus. En général, il s'agit des touches dites "contrôlées", c'est-à-dire frappées pendant que la touche **CONTROL** ou **CTRL** est appuyée.

Dans ce chapitre, nous figurerons la touche **CONTROL** par le signe ^ : pour la commande CONTROL + C, qui signifie "appuyez sur la touche **CONTROL** puis, sans la relâcher, sur la touche **C**" ; on écrira ^**C**.

LES COMMANDES DE DÉPLACEMENT DE CURSEUR

- ☐ ^**A** : déplacement d'un caractère à gauche ;
- ☐ ^**B** : déplacement en début de ligne ou en fin de ligne (passage successif en début et en fin de ligne) ;
- ☐ ^**F** : déplacement d'un caractère à droite ;
- ☐ ^**J** : déplacement ligne suivante ;
- ☐ ^**I** : déplacement d'une unité de tabulation (identique à la touche **TAB** si elle existe).

LES COMMANDES DE MODIFICATION D'EXÉCUTION

- ☐ ^**S** : suspend momentanément l'exécution d'un programme (STOP) ;
- ☐ ^**Q** : reprend l'exécution d'un programme, suspendue par ^S ;
- ☐ ^**C** : stoppe définitivement l'exécution d'un programme (BREAK) ;
- ☐ ^**P** : crée une double sortie des informations, une à l'écran, un double à l'imprimante. ^P est un bistable, c'est-à-dire qu'une nouvelle frappe libère la double sortie, etc.

LES COMMANDES D'ÉDITION EN COURS DE FRAPPE

☐ ^U : annule la ligne en cours de frappe ; un signe £ se met en bout de ligne, signalant qu'elle est annulée, et le curseur revient en début de ligne pour une nouvelle frappe ;

☐ ^X : efface le contenu de la ligne depuis le début jusqu'au curseur ;

☐ ^K : inverse de ^X, efface le contenu de la ligne depuis le curseur jusqu'à la fin de la ligne ;

☐ ^R : refrappe la ligne en cours de frappe (était autrefois utile lorsque l'on ne disposait que d'une imprimante comme périphérique de sortie ; on avait ainsi la ligne de commande "propre", sans rature) ;

☐ ^G : efface le caractère placé sous le curseur (identique à la touche **CLR** si elle existe) ;

☐ ^H : efface le caractère placé à gauche du curseur (identique à la touche **DEL** si elle existe).

COMMANDES DE L'ÉDITEUR DE COMMANDES

Cet éditeur permet de renouveler la dernière commande frappée.

☐ ^W : refrappe la ligne de commande entrée en dernier.

Les commandes d'édition en cours de frappe sont actives, ainsi que les commandes de déplacement du curseur à gauche et à droite.

La pratique de cet éditeur se révèle particulièrement efficace en gagnant un temps précieux, non seulement en cas d'erreur de frappe, mais aussi dans le cas de commandes multiples successives.

L'ÉDITEUR MINIMAL

Un petit utilitaire (associé à l'ordre PIP que nous verrons plus tard) permet de frapper un petit texte et de le stocker sur disque. Cet utilitaire est précieux dans le cas de frappe de petits textes, comme par exemple un fichier d'enchaînement d'ordres (SUBMIT).

☐ ^H : recule le curseur d'un caractère ;

☐ ^M : passe en début de ligne ;

☐ ^J : passe ligne suivante ;

☐ ^Z : marque la fin du texte.

Nous reviendrons sur l'utilisation de ces commandes dans le chapitre sur l'ordre PIP.

L'ORGANISATION DE LA DISQUETTE

LA FORME DES DISQUETTES

Il existe plusieurs types de disquettes caractérisées par leur diamètre et leur type de construction.

– Les disquettes à jaquettes souples ; les plus anciennes étaient de simples jaquettes souples contenant un disque magnétisable. Deux standards de ce type :

- le 8 pouces (environ 20 cm de diamètre, pratiquement abandonné aujourd'hui bien qu'il ait représenté le standard de stockage du CP/M 80) ;
- le 5 pouces 1/4 (environ 13 cm de diamètre, reste le standard des disquettes d'ordinateurs semi-professionnels).

– Les disquettes à jaquettes rigides ; de conception plus récente, elles cherchent à augmenter la sécurité du stockage en protégeant la disquette magnétisable sous une jaquette rigide et un volet occulteur qui la découvre uniquement dans le lecteur :

- le 3 pouces 1/2, le plus répandu ;
- le 3 pouces, nouveau venu sur le marché grand public.

L'ORGANISATION DES INFORMATIONS

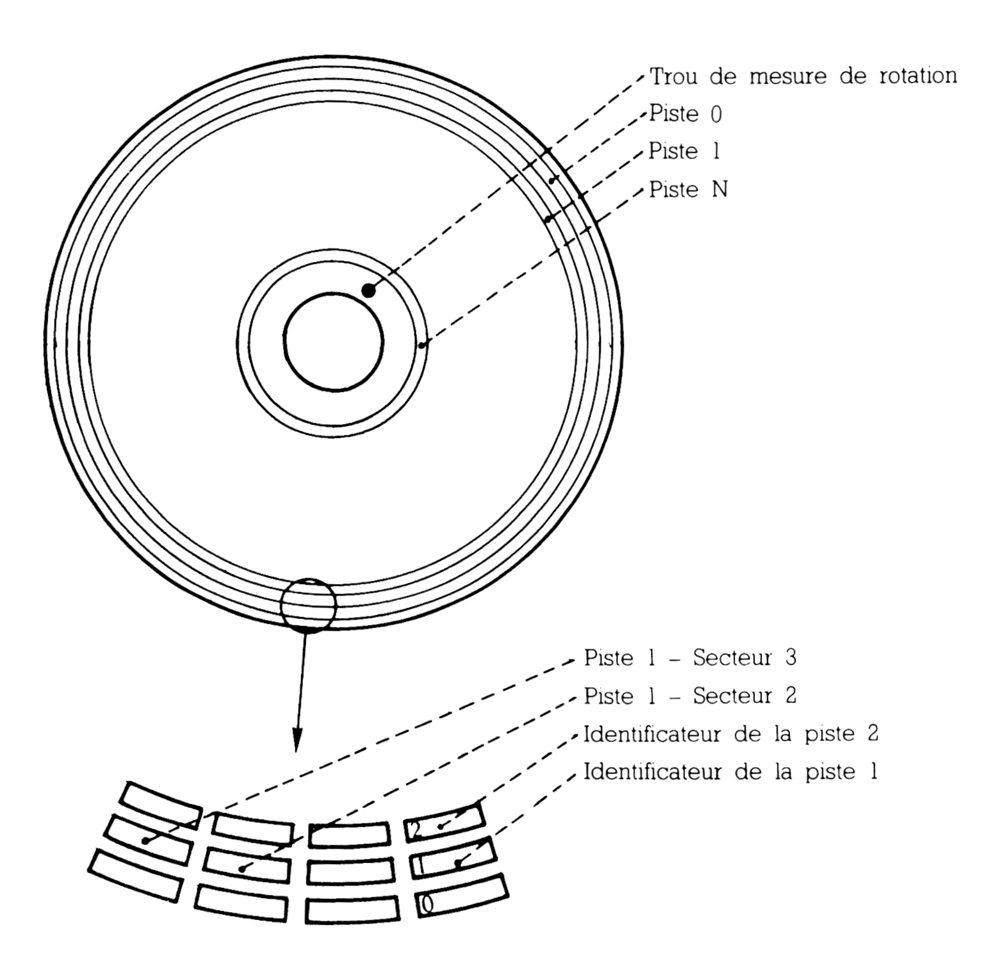
Une disquette gérée par CP/M devra être organisée de façon rigoureuse.
Les informations sont écrites sous forme de signaux magnétiques digitaux (une suite de signaux élémentaires qui, selon leur niveau, représenteront la valeur 0 ou la valeur 1).

Ces signaux sont groupés en octets (ensemble de huit signaux élémentaires).
Les octets sont groupés sur la disquette en secteurs.
Les secteurs sont groupés par pistes concentriques.
Les pistes sont groupées sur une face de disquette.

Certains lecteurs possèdent deux têtes de lecture, face à face, qui lisent les deux faces à la fois ; d'autres ont une seule tête ; pour accéder aux deux faces de la disquette, il est donc nécessaire de la retourner.

La largeur de la tête de lecture/écriture, associée à la qualité de l'électronique, caractérise la largeur des pistes et donc la densité radiale des informations. L'unité de mesure de cette densité est en « pistes par pouce » (Track Per Inch ou TPI) ; on parle de simple piste (48 TPI) ou de double piste (96 TPI), ou plus dans des lecteurs plus modernes.

Chaque bloc d'informations devra être identifié : au niveau de la disquette, des informations de structure précise devront signaler au système le début du secteur et son numéro, ainsi que le début et le numéro d'une piste.



Les données sont systématiquement organisées pour que CP/M puisse y accéder à partir d'un schéma de classement.

LE FORMATAGE

L'écriture de ces informations élémentaires mais indispensables pour le fonctionnement du système est l'opération de formatage. Cette opération, indispensable avant d'écrire sur une disquette, est effectuée par un utilitaire fourni par le constructeur de l'ordinateur.

LES ZONES USER

CP/M permet de définir 10 zones de la disquette appelées zones USER, caractérisées par leur numéro (de 0 à 9). Chaque zone est imperméable aux autres ; seuls les fichiers de la zone rendue active par l'ordre USER seront affectés par un ordre destructif. Les ordres de copie de fichiers seront actifs dans cette zone seule, sauf utilisation explicite d'une option de copie interzone (option G de l'ordre PIP).

Les fichiers affectés de l'attribut SYS résident physiquement dans une zone où ils peuvent être modifiés ou détruits. Cependant, ils sont utilisables de toute zone (cette facilité est utile pour les fichiers d'utilisation courante sous CP/M, en général tous les fichiers d'extension .COM).

UNITÉS LOGIQUES, UNITÉS PHYSIQUES

CP/M utilise comme périphériques d'entrée/sortie des unités logiques. Il s'agit de périphériques fictifs qui ne peuvent pas changer pour le système.

Les unités physiques sont par contre des périphériques de type bien défini : un ordre (DEVICE) permet donc d'attribuer et de changer l'attribution d'une unité physique à une unité logique.

En somme, CP/M adesse ses messages toujours à la même unité logique ; il appartient à l'utilisateur (ou au constructeur) d'indiquer au système quelles seront les unités réelles qui recevront le message adressé à telle ou telle unité logique.

LES UNITÉS LOGIQUES

☐ **CONIN** : entrée console. Clavier, le plus souvent ;

☐ **CONOUT** : sortie console. Écran, le plus souvent ;

☐ **AUXIN** : entrée auxiliaire. Voie série, le plus souvent, si elle existe ;

☐ **AUXOUT** : sortie auxiliaire. Voie série, également ;

☐ **LST** : imprimante parallèle, le plus souvent.

LES UNITÉS PHYSIQUES

☐ **CRT** : écran cathodique+clavier : c'est le classique terminal des gros ordinateurs, l'écran/clavier des micros ;

☐ **LPT** : imprimante rapide ;

☐ **SIO** : interface série.

D'autres noms d'unités sont possibles, dans l'ordre PIP en particulier ; nous les verrons à ce moment.

FONCTIONNEMENT AVEC UNE SEULE UNITÉ DE DISQUETTE

CP/M Plus sur l'Amstrad permet de travailler avec un seul disque comme avec plusieurs. Si l'on donne un ordre concernant un lecteur inexistant, CP/M conclut que l'on veut travailler sur une autre disquette : il demande, par un message défilant, de placer la nouvelle disquette.

Un ordre de copie d'un fichier du disque A vers B se fera donc sur le même lecteur, en changeant les disquettes à la demande du système.

LES ORDRES PRINCIPAUX EN DÉTAIL

INTRODUCTION

QUELQUES POINTS AVANT D'ENTRER DANS LE VIF DU SUJET

La syntaxe

Dans les questions simples que nous allons traiter, nous aborderons l'ordre nécessaire, la question qui se pose, la syntaxe pour la résoudre. Nous préciserons si l'ordre est résident ou s'il est vain d'essayer de faire fonctionner un ordre transitoire. Vérifiez l'existence du fichier nécessaire sur le disque :

Par exemple, pour lancer l'ordre DATE, assurez-vous de l'existence du fichier DATE.COM sur le disque, sinon la machine vous répondra DATE? ... Elle n'aura pas compris votre question, ne possédant pas les éléments pour y répondre.

Les types de questions

Nous avons essayé de traiter les questions les plus courantes qui se posent à l'utilisateur moyen de CP/M Plus. Certains ordres sont plus évolués, d'autres nous ont semblé trop complexes et trop exceptionnels d'utilisation pour les aborder ici. C'est notamment le cas des ordres liés à la programmation en Assembleur.

L'éditeur, ED, et l'outil de programmation avancée, SID, ne seront pas traités comme questions, mais comme deux petits chapitres à part.

Le truc

A la fin de certains chapitres, un petit paragraphe *Le truc* vous donnera certains conseils ou astuces sur l'utilisation particulière de l'ordre.

CP/M PLUS ET CP/M 2

Certaines machines sont livrées avec CP/M Plus (CP/M 3) et CP/M 2 ; les deux systèmes sont incompatibles entre eux à certains niveaux, mais vous aurez peut-être besoin d'utiliser l'un et l'autre. Nous notons donc en début de page CP/M Plus ou CP/M 2, selon l'appartenance exclusive de certains ordres à certains systèmes. Si aucune mention ne figure, cela signifie que l'ordre est commun aux deux systèmes.

INITIALISER LE SYSTÈME

COPYSYS

CP/M Plus

Copier le système CP/M sur un disque.

Syntaxe

COPYSYS

Ordre transitoire.

Dans le cas d'un disque nouvellement formaté ou ne comportant pas le système CP/M, l'ordre le copie sur le disque spécifié.

Exemple

```
A>COPYSYS

COPYSYS VER 3.0

SOURCE DRIVE NAME (OR RETURN FOR DEFAULT)?A

SOURCE ON  A  THEN TYPE RETURN

FUNCTION COMPLETE

DESTINATION DRIVE NAME (OR RETURN TO REBOOT)?B

DESTINATION ON  B  THEN TYPE RETURN

FONCTION COMPLETE

DO YOU WHISH TO COPY CPM.SYS? NO

A>_
```

L'ordre COPYSYS lance l'utilitaire qui fonctionne ensuite par dialogue ; il demande le disque source, le disque destination puis copie le système de base ; il demande ensuite s'il doit copier CPM.SYS (fichier des commandes) et l'effectue éventuellement en renouant le dialogue (source, destination...).

Le truc

Ordre indispensable dans le cas heureusement peu fréquent où le système est abîmé sur une disquette que l'on souhaite cependant réutiliser, il suffit parfois de régénérer le système pour que tout rentre dans l'ordre.

L'ordre COPYSYS ne formate pas la disquette, il ne joue que sur la piste système et n'affecte pas les autres.

SUBMIT

Lancer une suite d'ordres.

Syntaxe

SUBMIT fich

où fich est le fichier contenant la liste des ordres à effectuer.
Ordre transitoire.

Exemple

```
A>TYPE DEBUT.SUB
DATE SET
DATE
dir
DIRS

A>SUBMIT DEBUT

A>DATE SET

Enter today's date (MM/DD/YY): 11/17/85
Enter the time (HH:MM:SS):     20:15:00
Press any key to set time

A>DATE
Sun 11/17/85 20:15:02
A>dir
A: C10CPM3  EMS : DEBUT    SUB : SHOW     COM : SUBMIT   COM
A: SYSIN59  $$$ : DIR      COM : DATE     COM : PIP      COM
A>DIRS
NON-SYSTEM FILE(S) EXIST
A>
```

Le fichier DEBUT.SUB contient une suite d'ordres CP/M.
L'ordre SUBMIT DEBUT lance de façon automatique l'exécution de tous ces ordres.

Attention. – Pour être exécutable en procédure SUBMIT, le fichier appelé doit avoir l'extension .SUB.

Le truc

Préparez-vous pour chaque application un fichier SUBMIT (ou mieux, un fichier PROFILE.SUB (voir le chapitre Astuces) qui comprendra tous les ordres d'initialisation correspondant à votre application.

SUBMIT

Lancer une suite d'ordres avec des paramètres.

Syntaxe

SUBMIT fich par1 par2

où fich est le fichier SUBMIT ; par1 et par2 sont des paramètres utilisés par les ordres (on peut avoir 9 paramètres).

Ordre transitoire.

Exemple

```
A>TYPE LIST.SUB
dir *.$1
dir *.$2

A>SUBMIT LIST COM SUB

A>dir *.COM
A: SHOW     COM : SUBMIT   COM : DIR      COM : DATE     COM
A: PIP      COM
A>dir *.SUB
A: DEBUT    SUB : LIST     SUB
A>
```

Le fichier LIST.SUB contient des ordres CP/M (ici des DIR) ; ils sont suivis d'arguments et de numéros précédés du signe $.

Les signes $1 et $2 seront considérés comme la première variable et la seconde variable de l'ordre.

Les variables prennent les valeurs des mots placés après le nom du fichier dans l'ordre SUBMIT: le premier mot (COM, dans l'exemple, devient la première variable ; il remplace $1) ; le second mot (ici SUB) prend la place de la variable $2.

L'exécution se fait avec les variables remplacées.

DATE

CP/M Plus

Comment obtenir la date et l'heure.

Syntaxe

DATE

Ordre transitoire.

Les deux paramètres sont issus de l'horloge interne du système. Cette horloge n'est pas entretenue dans la plupart des machines grand public, ce qui signifie que les paramètres sont mis à zéro à la coupure de courant. De ce fait, il est nécessaire de faire la mise à l'heure à l'aide de l'ordre approprié en début de séquence de travail.

Exemple

```
A>DATE
Wed 12/15/82 00:20:03
A>
```

Le système répond à l'ordre DATE en donnant la date et l'heure.

La machine a calculé le jour de la semaine et l'affiche (en anglais, ici WED pour mercredi).

Si le système n'a pas été mis à l'heure et à la date, il affiche une date quelconque et le temps écoulé depuis la mise sous tension.

DATE

CP/M Plus

Afficher en permanence la date et l'heure.

Syntaxe

DATE C

Ordre transitoire.

Il est possible de laisser afficher la date et l'heure, celle-ci étant en permanence actualisée, pour garder une horloge sous les yeux, cela à l'aide de l'option C de l'ordre DATE.

Cet affichage est provisoire ; une nouvelle frappe au clavier le fait disparaître.

Exemple

```
A>DATE C
Wed 12/15/82 00:20:5[illegible]
```

On remarque le caractère gras de la ligne d'affichage, l'imprimante reproduisant l'affichage de l'écran qui réécrit sa ligne toutes les secondes. Les quatre derniers caractères de l'heure sont difficilement lisibles, puisque se superposent des chiffres différents.

DATE

CP/M Plus

Mettre à l'heure en une seule opération.

Syntaxe

DATE MM/JJ/AA HH:MM:SS

Ordre transitoire.

Il est possible de mettre l'horloge interne à l'heure en faisant suivre l'ordre DATE des paramètres à initialiser.

Exemple

```
A>DATE 10/09/85 11:11

ERROR: Illegal time/date specification.
A>

A>DATE 10/09/85 11:12:00

Strike key to set time

A>


A>DATE
Wed 10/09/85 11:12:06
A>
```

Dans la première opération, on a omis de donner les secondes au système : il répond en refusant de prendre l'initialisation. Les paramètres exhaustifs doivent suivre l'ordre DATE.

Il est prudent de donner une heure légèrement postérieure à l'heure précise d'initialisation, puisque le système demande de frapper une touche pour initialiser l'heure ; on peut donc regarder sa montre et initialiser à la seconde près si on le souhaite.

L'ordre DATE seul permet de vérifier éventuellement que les nouveaux paramètres ont bien été pris en compte.

DATE

CP/M Plus

Initialiser la date et l'heure par dialogue.

Syntaxe

DATE SET

Ordre transitoire.

Cet ordre est plus convivial et évite des erreurs de saisie, notamment en présentant la forme de la date et l'heure à saisir : cela limite les risques d'erreurs dues à la notation anglo-saxonne nécessaire au système et qui place le mois avant le jour.

Exemple

```
A>DATE SET

Enter today's date (MM/DD/YY): 10/09/85
Enter the time (HH:MM:SS):     11:12:30
Press any key to set time

A>
```

L'exemple se passe de commentaire ; une fois les paramètres saisis, on passe au dialogue de l'initialisation directe.

Le truc

La date et l'heure sont loin d'être des gadgets : si les options de datation automatique des fichiers ont été initialisées, il est souvent indispensable de connaître la chronologie de sauvegarde des fichiers.

L'ordre peut se simplifier en DATE S.

UTILISER LES PÉRIPHÉRIQUES

n:

Changer d'unité active de disquette.

Syntaxe

n:

ou

xn:

où n est le nom de l'unité que l'on veut rendre active ; il s'agit d'une lettre de A à P, selon le nombre d'unités connues du système.

x est le numéro de la zone utilisateur que l'on veut atteindre dans le disque considéré (chiffre de 0 à 9). Par défaut, c'est la zone 0 qui sera atteinte.

Exemple

```
A>DIR
A: C10CPM3  EMS : DIR      COM : ERASE    COM : PIP      COM
A: TEXTE1   TXT : SHOW     COM : TEXTE2   TXT : TEXTE3   TXT
A: INITDIR  COM : SET      COM : SETDEF   COM : TYPE     COM

A>B:
B>dir
B: C10CPM3  EMS : SHOW     COM : TEXTE    TXT : INITDIR  COM
B: DIR      COM : DATE     COM : PIP      COM : SET      COM
B: TYPE     COM

B>a:
A>

A>dir
A: C10CPM3  EMS : DIR      COM : ERASE    COM : PIP      COM
A: TEXTE1   TXT : SHOW     COM : TEXTE2   TXT : TEXTE3   TXT
A: INITDIR  COM : SET      COM : SETDEF   COM : TYPE     COM
2B:
2B>dir
B: ESS
2B>2A:
2A>dir
A: TEXTE    TXT
2A>USER 0
A>
```

Le premier directory est celui du disque A.

- **L'ordre B** : rend le B actif ; cela est confirmé par le prompt (B>) et par le contenu du directory.
- **L'ordre A** : rend le lecteur A actif ; l'ordre DIR confirme le retour.
- **L'ordre 2B** : rend le lecteur B actif, dans la zone USER 2, c'est ce que confirment le prompt (2B>) et le directory.
- **L'ordre 2A** : rend le lecteur A actif dans la zone 2.
- **L'ordre USER: 0** : replace la zone 0 du disque A active.

DEVICE

CP/M Plus

Lister les unités physiques connues de votre système.

Syntaxe

DEVICE NAME

Ordre transitoire.

Cet ordre liste à l'écran les périphériques connus du système et leur configuration.

Exemple

```
A>DEVICE NAME

Physical Devices:
I=Input,O=Output,S=Serial,X=Xon-Xoff
CRT     NONE  IO      LPT     NONE  O

A>
```

Nous voyons que seuls deux périphériques sont opérationnels : CRT (pour Cathode Ray Tube, par extension le terminal clavier/écran), qui est affecté en IO, c'est-à-dire en Input/Output, donc aussi bien en entrée d'informations (clavier) qu'en sortie (écran), et LPT (pour Line PrinTer), qui est affecté en sortie seulement. L'information NONE indique que les périphériques sont en liaison parallèle. (S'ils étaient en série, il y aurait l'indication de leur vitesse de transmission).

Si l'ordinateur était équipé d'une interface série, l'unité SIO figurerait. On aurait en outre ses options (S pour série, éventuellement X si le protocole de dialogue de transmissions est activé).

Le truc

L'ordre abrégé :

DEVICE N

DEVICE
CP/M Plus

Lister les unités assignées.

Syntaxe

DEVICE VALUES

Ordre transitoire.

Cet ordre sert à visualiser l'affectation des unités physiques aux unités logiques du système.

Exemple

```
A>DEVICE VALUES

Current Assignments:
CONIN:   = CRT
CONOUT:  = CRT
AUXIN:   = Null Device
AUXOUT:  = Null Device
LST:     = LPT


A>
```

On voit que le terminal écran/clavier est affecté aux entrées/sorties console (CONIN: et CONOUT:) ; que les entrées/sorties auxiliaires ne sont pas affectées, ce qui est normal compte tenu de l'absence d'interface série dans le système utilisé ; enfin que la sortie de listage est, quant à elle, affectée à LPT, c'est-à-dire à l'imprimante parallèle.

Le truc

L'ordre abrégé :

DEVICE V

DEVICE
CP/M Plus

Obtenir la liste complète des périphériques.

Syntaxe

DEVICE

Ordre transitoire.

L'ordre sans argument donne la liste et les affectations des périphériques ; il permet en outre de réaffecter une unité physique.

Exemple

```
A>DEVICE

Physical Devices:
I=Input,O=Output,S=Serial,X=Xon-Xoff
CRT     NONE  IO     LPT     NONE  O

Current Assignments:
CONIN:   = CRT
CONOUT:  = CRT
AUXIN:   = Null Device
AUXOUT:  = Null Device
LST:     = LPT

Enter new assignment or hit RETURN


A>
```

On voit la juxtaposition des réponses aux ordres DEVICE NAME et DEVICE VALUES et, en fin de réponse, une ligne de proposition de réaffectation.

Le fait de presser la touche **RETURN** repasse le contrôle à CP/M sans changer l'affectation des périphériques.

Le truc

Possibilité bien intéressante lorsque l'on a des anomalies de fonctionnement et qu'il faut d'abord vérifier l'état des affectations : un ordre fréquemment utilisé par les amateurs avertis et les spécialistes de la communication.

DEVICE
CP/M Plus

Obtenir l'état logique ou physique de l'écran/clavier.

Syntaxe

DEVICE CON
ou DEVICE CRT

Ordre transitoire.

Ces ordres servent à visualiser avec précision l'état de l'écran/clavier qui reste le périphérique de visualisation privilégié du micro-ordinateur.

Exemple

```
A>DEVICE CON
CONIN:  = CRT
CONOUT: = CRT


A>
```

On voit que l'entrée console (CONIN:) et la sortie (CONOUT:) sont affectées à l'unité physique CRT, donc à l'écran/clavier.

Exemple

```
A>DEVICE CRT

Physical Device:  CRT
Baud Rate:        NONE
Characteristics:  INPUT
                  OUTPUT
                  PARALLEL

A>
```

L'écran/clavier a les caractéristiques suivantes : pas de vitesse de transfert (réservé à la voie série), il travaille en entrée, en sortie et sous un mode de transmission parallèle.

DEVICE
CP/M Plus

Réassigner une unité logique à une unité physique.

Syntaxe

DEVICE log:=phys

Ordre transitoire.

Il est possible de réassigner un périphérique à une unité logique. Cette opération s'avérera nécessaire pour utiliser une imprimante série, par exemple.

Exemple

```
A>DEVICE CONOUT:=LPT

PPhhyyssiiccaall  DDeevviicceess::

II==IInnppuutt,,OO==OOuuttppuutt,,SS==SSeer

CCRRTT          NNOONNEE    IIOO           LLP


CCuurrrreenntt  AAssssiiggnnmmeennttss::

CCOONNIINN::     ==  CCRRTT

CCOONNOOUUTT::  ==  LLPPTT

AAUUXXIINN::     ==  NNuullll  DDeevviiccee

AAUUXXOOUUTT::  ==  NNuullll  DDeevviiccee

LLSSTT::        ==  LLPPTT


AA>>DDEEVVIICCEE  CCOONNOOUUTT::==CCRRTT


Physical Devices:
I=Input,O=Output,S=Serial,X=Xon-Xoff
CRT    NONE  IO     LPT    NONE  O

Current Assignments:
CONIN:  = CRT
CONOUT: = CRT
AUXIN:  = Null Device
AUXOUT: = Null Device
LST:    = LPT


A>
```

Dès la prise en compte de la réassignation, la machine donne pour mémoire l'état des affectations. On remarque que toutes les lettres sont doublées puisque, le bistable ^P étant actif, on a déjà une première copie des lettres vers l'imprimante.

L'ordre DEVICE CONOUT:=CRT rétablit l'état initial.

Le truc

Cette possibilité sera parfois utilisée en procédure automatique pour dérouter une sortie écran vers l'imprimante, dans le cas de la réception d'un dialogue entre ordinateurs par la voie série, par exemple.

DEVICE
CP/M Plus

Quelles sont les caractéristiques de l'affichage écran ?

Syntaxe

DEVICE CONSOLE [PAGE]

Ordre transitoire.
L'ordre DEVICE permet de savoir quelles sont les caractéristiques de l'affichage alphanumérique, c'est-à-dire le nombre de lignes et le nombre de colonnes.

Exemple

```
A>DEVICE CONSOLE[PAGE]

Console width set to 79 columns
Console page set to 24 lines


A>
```

La réponse est de 79 colonnes par 24 lignes ; cette valeur est la valeur prise par défaut par le système, faute de précision sur la taille.

Le truc

Ordre abrégé :

DEVICE C[P]

DEVICE
CP/M Plus

Modifier les caractéristiques de l'affichage écran.

Syntaxe

DEVICE CONSOLE [COLUMNS=xx LINES=yy]

Ordre transitoire.
Il est possible de modifier les caractéristiques d'affichage pour visualiser selon des nombres de lignes et de colonnes différents.

Exemple

```
A>DEVICE CONSOLE[COLUMNS=10 LINES=10]

Console width set to 10 columns
Console page set to 10 lines


A>DEVICE
CONSOLE[C
OLUMNS=79
 LINES=24
]

Console width set to 79 columns
Console page set to 24 lines
```

L'ordre DEVICE CONSOLE [COLUMNS= 10 LINES=10] modifie les caractéristiques d'affichage.

Ces caractéristiques apparaissent nettement à la frappe de l'ordre de remise à l'état initial de 79 × 24.

A chaque utilisation de cet ordre, la machine répond en signalant les nouvelles valeurs qu'elle a prises en compte.

Le truc

Cette possibilité est parfois utilisée pour visualiser à l'écran la césure de lignes d'une imprimante de 132 colonnes ou l'état d'une page de texte qui devra passer sur un autre type d'écran (vidéotexte, par exemple).

Un ordre abrégé (ouf !) :

DEVICE C[C=xx L=yy]

LA LISTE DES FICHIERS

DIR

Afficher la liste simplifiée d'une disquette.

Syntaxe

DIR

Ordre résident.
Cet ordre élémentaire permet d'obtenir la liste simple des fichiers.

Exemple

```
A>dir
A: C10CPM3  EMS : PROFILE  ENG : SUBMIT   COM : DISCKIT3 COM
A: DEVICE   COM : DIR      COM : ERASE    COM : GET      COM
A: PIP      COM : PUT      COM : RENAME   COM : SHOW     COM
A: TYPE     COM : SET      COM
SYSTEM FILE(S) EXIST
A>
```

La liste des fichiers est donnée dans l'ordre du directory (sans classement alphabétique).

Cette liste sur quatre colonnes ne donne que le nom complet des fichiers (nom du disque, nom du fichier, nom de l'extension).

La mention "SYSTEM FILE(S) EXIST" signale que le directory renferme des fichiers d'attribut SYS qui n'apparaissent pas au directory simple, mais à l'ordre DIRS.

DIR

Afficher le directory d'un autre disque.

Syntaxe

DIR c:

où c est le nom du lecteur à explorer.

Ordre résident.

Exemple

```
A>DIR B:
B: C10CPM3  EMS : PIP      COM : TEXTE2   TXT : TEXTE    TXT
B: TEXTE1   TXT : COPYR
A>
```

Le "prompt", ou signal d'attente du système (A>), signale que le disque A est actif ; l'ordre DIR B: demande au système d'aller lire le lecteur B ; les noms des fichiers listés au directory sont bien précédés de la lettre B, signalant qu'ils se trouvent sur le lecteur B.

En fin d'exécution de l'ordre, le prompt (A>) signale que le lecteur A est toujours le lecteur actif.

DIR

Afficher la liste des fichiers d'une extension particulière.

Syntaxe

DIR *.EXT

Ordre résident.

Cet ordre permet de lister une partie du disque selon des similitudes au niveau de l'extension ; c'est une illustration de l'utilisation des caractères d'ambiguïté.

Exemple

```
A>dir *.com
A: SUBMIT   COM : DISCKIT3 COM : DATE     COM : DEVICE   COM
A: DIR      COM : ERASE    COM : GET      COM : PIP      COM
A: PUT      COM : RENAME   COM : SHOW     COM : TYPE     COM
A: SET      COM : AMSDOS   COM
A>
```

Seuls les fichiers présentant l'extension COM sont listés.

Le truc

Cette possibilité est souvent utilisée sur des disquettes communes à plusieurs personnes ; une extension portant les initiales de chaque programmeur permet de ne lister que ses propres fichiers.

DIR

Afficher la liste des fichiers ayant le même début de nom.

Syntaxe

DIR N*.*

Ordre résident.

Cet ordre élémentaire permet d'obtenir la liste simple des fichiers dont le début du nom correspond à la demande.

Exemple

```
A>DIR D*.*
A: DISCKIT3 COM : DATE     COM : DEVICE   COM : DIR      COM
A>
```

On trouve la liste des fichiers dont le nom commence par la lettre D.

Le truc

On utilise parfois cette possibilité pour sauvegarder des versions successives d'un même programme que l'on développe ; par exemple, le programme de nom PROG aura les versions successives PROGO1, PROGO2, PROGO3 ... Il suffira de demander la liste par l'ordre PROG*.* pour avoir la liste de tous les fichiers, de l'application et, éventuellement, (avec des options à l'ordre) leur date de création, leur taille, etc.

DIR
CP/M Plus

Afficher les attributs des fichiers.

Syntaxe

DIR [ATT]

Ordre transitoire.

Cet ordre donne la liste triée par ordre alphabétique des fichiers, avec d'autres renseignements dont leur taille, le nombre de blocs et les attributs.

Cet ordre est identique, dans ses résultats, à l'ordre

DIR [FULL]

Exemple

```
A>DIR [ATT]

Scanning Directory...

Sorting  Directory...

Directory For Drive A:  User  0

    Name      Bytes   Recs   Attributes       Name      Bytes   Recs   Attributes
------------ ------ ------ ------------  ------------ ------ ------ ------------
AMSDOS   COM     1k      8 Dir RW        C10CPM3  EMS    25k    200 Dir RW
DATE     COM     3k     23 Dir RW        DEVICE   COM     8k     58 Dir RW
DIR      COM    15k    114 Dir RW        DISCKIT3 COM     6k     48 Dir RW
ERASE    COM     4k     29 Dir RW        GET      COM     7k     51 Dir RW
PIP      COM     9k     68 Dir RW        PROFILE  ENG     1k      1 Dir RW
PUT      COM     7k     55 Dir RW        RENAME   COM     3k     23 Dir RW
SET      COM    11k     81 Dir RW        SHOW     COM     9k     66 Dir RW
SUBMIT   COM     6k     42 Dir RW        TYPE     COM     3k     24 Dir RW

Total Bytes     =    118k  Total Records =     891  Files Found =    16
Total 1k Blocks =    118   Used/Max Dir Entries For Drive A:    18/  64

A>
```

On voit le résultat de l'ordre qui délivre de nombreux renseignements. Les deux dernières lignes donnent des informations générales sur les fichiers : leur nombre (ici 16), leur taille totale (115 K octets), le nombre de secteurs, le nombre de blocs, d'entrées directory...

Le truc

L'abréviation DIR [A] est parfaitement opérationnelle.

A la suite d'une option, il est possible de placer des ambiguïtés de noms de fichiers (DIR [ATT] *.COM *.TXT donnera la liste avec attributs des fichiers d'extension COM et TXT).

Attention. – Cet ordre, ainsi que tous les ordres DIR qui vont suivre, sont des ordres de listage de directory avec option. Ils peuvent tous comporter un nom de fichier avant l'option ou un nom de fichier ambigu ; dans ce cas, l'ordre s'appliquera à une liste partielle correspondant au nom de fichier exprimé.

DIR
CP/M Plus

Lister les fichiers avec date de création.

Syntaxe

DIR [FULL]

Ordre transitoire.

Exemple

```
dir [full]

Scanning Directory...

Sorting  Directory...

Directory For Drive A:  User  0

    Name      Bytes   Recs   Attributes   Prot      Update           Access
------------ ------ ------ ------------ ------ --------------   --------------

C10CPM3  EMS    25k    200 Dir RW       None
DIR      COM    15k    114 Dir RW       None                     12/15/82 01:59
SHOW     COM     9k     66 Dir RW       None                     12/15/82 00:15
TEXTE1   TXT     1k      1 Dir RW       None                     12/15/82 00:00
TEXTE2   TXT     1k      1 Dir RW       Read                     12/15/82 00:21
TEXTE3   TXT     1k      1 Dir RW       Read                     12/15/82 00:17

Total Bytes     =     52k  Total Records =      383  Files Found =    6
Total 1k Blocks =     52   Used/Max Dir Entries For Drive A:   19/  64

A>
```

L'affichage du directory étendu se fait automatiquement avec ce format si le disque a été initialisé pour la datation (ordre INITDIR).

DIR

Lister les fichiers selon leur attribut DIR ou SYS.

Syntaxe

DIR

ou DIRS

Ordres résidents.

L'ordre DIR liste les fichiers d'attribut DIR, l'ordre DIRS liste les fichiers d'attribut SYS.

L'ordre SET permet de changer les attributs.

Exemple

```
A>DIR
A: C10CPM3   EMS : PROFILE  ENG : SUBMIT   COM : DISCKIT3 COM
A: DEVICE    COM : DIR      COM : ERASE    COM : GET      COM
A: PIP       COM : PUT      COM : RENAME   COM : SHOW     COM
A: TYPE      COM
SYSTEM FILE(S) EXIST
A>

A>DIRS
A: SET       COM : AMSDOS   COM
NON-SYSTEM FILE(S) EXIST
A>
```

La première liste fait apparaître les fichiers d'attribut DIR, le système signale qu'il existe un ou des fichiers d'attribut SYS.

La seconde liste donne les fichiers d'attribut SYS en signalant l'existence de fichiers d'attribut DIR.

Attention. – L'ordre DIRS ne fonctionne pas avec CP/M 2, mais uniquement avec CP/M 86 et CP/M Plus.

Le truc

Si vous voulez avoir les mêmes listes, mais triées par ordre alphabétique et renfermant plus de renseignements sur les fichiers, les ordres transitoires DIR [DIR] et DIR [SYS] vous rendront ce service.

DIR

CP/M Plus

Lister les fichiers de tous les lecteurs actifs.

Syntaxe

DIR [DRIVE=ALL]

Ordre transitoire.

Cet ordre permet de connaître, en une seule exploration, le contenu de tous les lecteurs actifs.

Exemple

```
A>DIR[drive=all]

Scanning Directory...

Sorting  Directory...

Directory For Drive A:  User  0

    Name     Bytes   Recs   Attributes       Name     Bytes   Recs   Attributes
------------ ------ ------ ------------ ------------ ------ ------ ------------
C10CPM3  EMS    25k    200 Dir RW       COPYR           1k      1 Dir RW
DIR      COM    15k    114 Dir RW       PIP      COM    9k     68 Dir RW
TEXTE    TXT     1k      1 Dir RW       TEXTE1   TXT    1k      1 Dir RW
TEXTE2   TXT     1k      2 Dir RW

Total Bytes     =     53k  Total Records =     387  Files Found =     7
Total 1k Blocks =     53   Used/Max Dir Entries For Drive A:    9/  64

Scanning Directory...

Sorting  Directory...


Directory For Drive B:  User  0

    Name     Bytes   Recs   Attributes       Name     Bytes   Recs   Attributes
------------ ------ ------ ------------ ------------ ------ ------ ------------
C10CPM3  EMS    25k    200 Dir RW       ESS      TXT    1k      1 Dir RW
PIP      COM     9k     68 Dir RW       TEXTE    TXT    1k      1 Dir RW

Total Bytes     =     36k  Total Records =     270  Files Found =     4
Total 1k Blocks =     36   Used/Max Dir Entries For Drive B:    9/  64

A>
```

Deux lecteurs sont actifs ; le système répond en donnant, sous forme étendue, la liste de tous les fichiers des deux disques.

L'option DRIVE peut avoir un ou plusieurs noms en argument (*exemples* : DRIVE=A,B ou DRIVE=A,B,D), pourvu que le système les reconnaisse.

L'option DRIVE peut se combiner avec l'option USER pour donner les contenus des zones utilisateur. Par défaut, les zones USER prises en compte sont les zones actives de chaque disque. L'ordre DIR [DRIVE=ALL USER=ALL] donnera le contenu **total** de tous les lecteurs connus du système.

DIR

Lister les fichiers avec tous leurs paramètres.

Syntaxe

DIR [ALL]

Ordre transitoire.

Cet ordre, identique à DIR [FULL] et semblable à DIR [ATT] dans ses effets, permet une liste complète des fichiers, triés par ordre alphabétique des noms et donnant leur taille, le nombre d'enregistrements de 128 octets, et les attributs.

Une totalisation d'encombrement des fichiers est faite en fin de liste.

Exemple

```
A>dir [FULL]

Scanning Directory...

Sorting  Directory...

Directory For Drive A:  User  0

    Name       Bytes   Recs   Attributes       Name       Bytes   Recs   Attributes
------------ ------ ------ ------------   ------------ ------ ------ -----------
AMSDOS   COM     1k      8 Dir RW          C10CPM3  EMS    25k    200 Dir RW
DATE     COM     3k     23 Dir RW          DEVICE   COM     8k     58 Dir RW
DIR      COM    15k    114 Dir RW          DISCKIT3 COM     6k     48 Dir RW
ERASE    COM     4k     29 Dir RW          GET      COM     7k     51 Dir RW
PIP      COM     9k     68 Dir RW          PROFILE  ENG     1k      1 Dir RW
PUT      COM     7k     55 Dir RW          RENAME   COM     3k     23 Dir RW
SET      COM    11k     81 Dir RW          SHOW     COM     9k     66 Dir RW
SUBMIT   COM     6k     42 Dir RW          TYPE     COM     3k     24 Dir RW

Total Bytes     =    118k  Total Records =     891  Files Found =    16
Total 1k Blocks =    118   Used/Max Dir Entries For Drive A:   18/   64

A>
```

La liste est donc bien identique à celle de l'ordre DIR [ATT] que nous avons vue précédemment.

Le truc

Là encore, DIR [A] aura le même effet et sera plus rapide à frapper au clavier.

DIR

CP/M Plus

Lister les fichiers à l'exclusion de certains.

Syntaxe

DIR fich?* [EXCLUDE]

Ordre transitoire.

La liste qui sera affichée sera celle des fichiers qui ne correspondent pas au nom de fichier spécifié ou à l'ambiguïté qui en tient lieu.

Exemple

```
A>DIR *.COM [EXCLUDE]

Scanning Directory...

Sorting  Directory...

Directory For Drive A:  User  0

    Name      Bytes   Recs   Attributes      Name      Bytes   Recs   Attributes
------------ ------ ------ ------------ ------------ ------ ------ ------------
C10CPM3  EMS    25k     200 Dir RW       PROFILE  ENG     1k       1 Dir RW

Total Bytes     =     26k  Total Records =      201  Files Found =    2
Total 1k Blocks =     26   Used/Max Dir Entries For Drive A:   18/  64

A>
```

Les fichiers listés sur le modèle de tous les directory avec option (par ordre alphabétique, avec encombrement et attributs) sont ceux qui n'ont pas l'extension .COM puisque le nom de fichier à exclure figurant dans l'ordre est *.COM.

Le truc

L'option peut être simplifiée par l'abréviation DIR [E] qui fonctionne de façon identique.

DIR

CP/M Plus

Lister les fichiers dans l'ordre du directory.

Syntaxe

DIR [NOSORT]

Ordre transitoire.

Cet option permet d'obtenir une liste non triée dans l'ordre alphabétique, donc dans l'ordre d'inscription au directory.

Exemple

```
A>DIR [NOSORT]

Scanning Directory...

Directory For Drive A:  User  0

    Name      Bytes   Recs   Attributes      Name      Bytes   Recs   Attributes
------------ ------ ------ ------------ ------------ ------ ------ ------------
C10CPM3  EMS    25k     200 Dir RW       PROFILE  ENG     1k       1 Dir RW
SUBMIT   COM     6k      42 Dir RW       DISCKIT3 COM     6k      48 Dir RW
DATE     COM     3k      23 Dir RW       DEVICE   COM     8k      58 Dir RW
DIR      COM    15k     114 Dir RW       ERASE    COM     4k      29 Dir RW
GET      COM     7k      51 Dir RW       PIP      COM     9k      68 Dir RW
PUT      COM     7k      55 Dir RW       RENAME   COM     3k      23 Dir RW
SHOW     COM     9k      66 Dir RW       TYPE     COM     3k      24 Dir RW
SET      COM    11k      81 Dir RW       AMSDOS   COM     1k       8 Dir RW

Total Bytes     =    118k  Total Records =      891  Files Found =   16
Total 1k Blocks =    118   Used/Max Dir Entries For Drive A:   18/  64

A>
```

Le truc

L'ordre peut se simplifier en DIR [NOS].

DIR
CP/M Plus

Lister les fichiers protégés (ou non protégés).

Syntaxe

DIR [RW]

ou DIR [RO]

Ordres transitoires.

Les fichiers listés seront ceux qui auront l'attribut correspondant : RO pour lecture seulement, RW pour lecture ou écriture. Ces attributs de protection sont placés par l'ordre SET.

Exemple

```
A>dir [rw]

Scanning Directory...

Sorting  Directory...

Directory For Drive A:  User  0

    Name      Bytes   Recs   Attributes      Name      Bytes   Recs   Attributes
------------ ------ ------ ------------ ------------ ------ ------ ------------
C1OCPM3  EMS    25k    200 Dir RW       DATE     COM     3k     23 Dir RW
DEVICE   COM     8k     58 Dir RW       DIR      COM    15k    114 Dir RW
DISCKIT3 COM     6k     48 Dir RW       PIP      COM     9k     68 Dir RW
PROFILE  ENG     1k      1 Dir RW       PUT      COM     7k     55 Dir RW

Total Bytes     =     74k  Total Records =     567  Files Found =    8
Total 1k Blocks =     74   Used/Max Dir Entries For Drive A:   18/  64

A>
```

La liste obtenue se fait, bien sûr, selon le format de toutes les listes avec options.

DIR
CP/M Plus

Obtenir la liste des fichiers avec leur taille.

Syntaxe

DIR [SIZE]

Ordre transitoire.

La forme de l'affichage sera modifiée par l'option ; seuls figureront les noms de fichiers et leur taille ; l'extension et le nombre de blocs ne seront pas présents dans la liste.

Exemple

```
A>dir [size]

Scanning Directory...

Sorting  Directory...

Directory For Drive A:  User  0

A: AMSDOS   COM    1k : C10CPM3  EMS   25k : DATE     COM    3k
A: DEVICE   COM    8k : DIR      COM   15k : DISCKIT3 COM    6k
A: ERASE    COM    4k : GET      COM    7k : PIP      COM    9k
A: PROFILE  ENG    1k : PUT      COM    7k : RENAME   COM    3k
A: SET      COM   11k : SHOW     COM    9k : SUBMIT   COM    6k
A: TYPE     COM    3k

Total Bytes     =    118k  Total Records =     891  Files Found =   16
Total 1k Blocks =    118   Used/Max Dir Entries For Drive A:   18/  64
```

Le récapitulatif en fin de ligne est identique aux autres options ; seuls les renseignements pour chaque fichier sont réduits.

Le truc

Ordre abrégé :

DIR [SI]

DIR
CP/M Plus

Lister les fichiers de toutes les zones utilisateurs.

Syntaxe

DIR [USER=ALL]

Ordre transitoire.

L'option USER=ALL permet d'obtenir le directory de la disquette complète, y compris le contenu des zones USER non actives.

Exemple

```
A>DIR [USER=ALL]

Scanning Directory...

Sorting  Directory...

Directory For Drive A:  User  0

    Name       Bytes   Recs   Attributes      Name       Bytes   Recs   Attributes
------------ ------ ------ ------------ ------------ ------ ------ -----------
C10CPM3  EMS    25k     200 Dir RW       DIR      COM    15k     114 Dir RW
SET      COM    11k      81 Dir RW

Total Bytes     =     51k  Total Records =     395  Files Found =    3
Total 1k Blocks =     51   Used/Max Dir Entries For Drive A:    7/  64


Directory For Drive A:  User  1

    Name       Bytes   Recs   Attributes      Name       Bytes   Recs   Attributes
------------ ------ ------ ------------ ------------ ------ ------ -----------
PIP      COM     9k      68 Dir RW       TYPE     COM     3k      24 Dir RO

Total Bytes     =     12k  Total Records =      92  Files Found =    2
Total 1k Blocks =     12   Used/Max Dir Entries For Drive A:    7/  64
```

Deux zones USER sont utilisées, leur contenu est listé.

Le truc

Ordre abrégé :

DIR [U=AL]

Il est possible de n'obtenir le directory que d'une seule zone USER en donnant son numéro au lieu de l'option ALL.

Exemple : DIR [USER=2] donnera le directory de la zone USER numéro 2.

DIR

CP/M Plus

Combiner les options de l'ordre DIR.

Syntaxe

DIR [opt1 opt2...]

Ordre transitoire.

Les options sont combinables. Elles doivent être séparées par un espace et être toutes comprises entre les deux crochets de contenu d'option.

Exemple

```
A>dir [nosort ro]

Scanning Directory...

Directory For Drive A:  User  0

    Name      Bytes   Recs   Attributes      Name      Bytes   Recs   Attributes
------------ ------ ------ ------------ ------------ ------ ------ ------------
SUBMIT   COM     6k     42 Dir RO       ERASE    COM     4k     29 Dir RO
GET      COM     7k     51 Dir RO       RENAME   COM     3k     23 Dir RO
SHOW     COM     9k     66 Dir RO       TYPE     COM     3k     24 Dir RO
SET      COM    11k     81 Dir RO       AMSDOS   COM     1k      8 Dir RO

Total Bytes     =     44k  Total Records =     324  Files Found =     8
Total 1k Blocks =     44   Used/Max Dir Entries For Drive A:   18/   64

A>
```

On a combiné l'option de sortie non classée et celle de recherche des fichiers en lecture seule ; le directory en sortie est donc non classé et ne contient que les fichiers d'option RO.

INITDIR

CP/M Plus

Initialiser un disque pour dater les fichiers.

Syntaxe

INITDIR n:

où n est le nom du lecteur où se trouve le disque à initialiser.

Cet ordre modifie la taille du directory pour permettre de dater les fichiers. Il ne modifie pas les données de la disquette.

Ordre transitoire.

Exemple

```
B>INITDIR
ERROR:  Unrecognized drive.
DRIVE:

Enter Drive: B


INITDIR WILL ACTIVATE TIME STAMPS FOR SPECIFIED DRIVE.
Do you want to re-format the directory on drive: B  (Y/N)?  Y


B>dir [full]

Scanning Directory...

Sorting  Directory...

Directory For Drive B:  User  0

    Name       Bytes   Recs   Attributes   Prot      Update          Create
------------ ------ ------ ------------ ------ -------------- --------------

C1OCPM3  EMS    25k    200 Dir RW       None
DATE     COM     3k     23 Dir RW       None
DIR      COM    15k    114 Dir RW       None
```

L'ordre donné sans argument ne peut pas être traité par le système ; il demande alors de lui préciser le nom du lecteur à initialiser.

L'activation se fait en reformatant le directory pour laisser la place de datation.

Un directory étendu confirme le nouveau format qui tient compte des dates de création et de mise à jour.

Attention. – Cet ordre lancé sur un disque déjà formaté pour la datation annule le format de datation (après demande de confirmation).

Le truc

Prévoyez de compléter votre environnement avec l'ordre DATE SET au début de chaque séquence, et avec l'ordre SET ACCESS, UPDATE ou CREATE pour préciser le type de mise à jour à effectuer.

OPÉRATIONS ÉLÉMENTAIRES SUR LES FICHIERS

TYPE

Lister le contenu d'un fichier texte.

Syntaxe

TYPE fich

Ordre résident.

L'ordre TYPE permet d'afficher à l'écran le contenu d'un fichier. Seuls les caractères affichables seront lisibles ; les caractères binaires (opérateurs machine d'un programme) seront affichés de manière incompréhensible.

Le nom du fichier ne doit pas contenir de caractère d'ambiguïté.

Exemple

```
A>TYPE CPM.COP
COPYRIGHT (C) 1975, 1976, 1977, 1978
;        DIGITAL RESEARCH
;        BOX 579, PACIFIC GROVE
;        CALIFORNIA, 93950

A>
```

Le petit fichier CPM.COP qui contient le texte du copyright de CP/M est affiché à l'écran.

Le truc

L'ordre TYPE est utilisable avec l'option PAGE. (CP/M Plus uniquement). Par exemple,

TYPE fichier [PAGE]

affiche le contenu du fichier de nom "fichier" par blocs de 24 lignes, en invitant à appuyer sur une touche pour visualiser les 24 autres lignes. Cette option est bien utile dans le cas de fichiers très longs.

DUMP

Lister le contenu d'un fichier en hexadécimal.

Syntaxe

DUMP fich

Ordre transitoire.

Cet ordre permet l'affichage d'un fichier en hexadécimal. Cela peut présenter un intérêt pour examiner certains programmes.

Exemple

```
A>DUMP CPM.COP

CP/M 3 DUMP - Version 3.0
0000: 43 50 2F 4D 20 56 65 72 73 69 6F 6E 20 33 2E 30  CP/M Version 3.0
0010: 43 4F 50 59 52 49 47 48 54 20 31 39 38 32 2C 20  COPYRIGHT 1982,
0020: 44 49 47 49 54 41 4C 20 52 45 53 45 41 52 43 48  DIGITAL RESEARCH
0030: 0D 0A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A  ................
0040: 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A  ................
0050: 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A  ................
0060: 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A  ................
0070: 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A 1A  ................
A>
```

On voit que DUMP ne se contente pas de lister en hexadécimal ; il donne les adresses hexa où se trouve l'octet et affiche à droite la traduction ASCII de cet octet lorsqu'elle est possible. Ici, on voit que le C de Digital Research se trouve à l'adresse 2E et qu'il est représenté par la valeur 43 hexa, c'est-à-dire C en code ASCII.

ERASE

Effacer un fichier.

Syntaxe

ERA fich

ou ERASE fich

Ordre résident.
L'effacement de fichier se fait sur un nom explicite ou sur un nom comportant des ambiguïtés ; dans ce cas, plusieurs fichiers sont détruits.

Exemple

1. -
```
A>DIR
A: RENAME   COM : PIP      COM : ERASE   COM

A>ERASE RENAME.COM

A>dir
A: PIP      COM : ERASE    COM
```

2. -
```
A>ERASE
Enter filename: PIP.COM
A: PIP     .COM (Y/N)? y

A>dir
A: ERASE    COM
A>
```

1. - Effacement sur ordre résident

L'ordre ERASE RENAME.COM (qui aurait pu s'écrire ERA RENAME.COM) a effacé le fichier nommé, comme on le voit dans le directory suivant.

2. - Effacement sur ordre transitoire (CP/M Plus seulement)

Si le fichier ERASE.COM se trouve sur la disquette, l'ordre ERASE sans argument lance un dialogue d'effacement tel qu'il est illustré dans l'exemple.

ERASE

Effacer certains fichiers sur ambiguïté de nom.

Syntaxe

ERASE fich?*

Ordre résident.
L'ordre ERASE (ou ERA) accepte des ambiguïtés dans le nom argument ; il est donc possible d'effacer plusieurs fichiers ayant des similitudes dans leur nom.

Exemple

```
A>DIR
A: RENAME   COM : TEXTE1   TXT : TXT2     TXT : PIP      COM
A: TXT3     TXT : ERASE    COM
A>

A>era *.TXT
ERASE *.TXT (Y/N)? Y


A>DIR
A: RENAME   COM : PIP      COM : ERASE    COM
A>
```

On voit que l'effacement se fait après une demande de confirmation. Cette demande est systématique dès qu'il y a possibilité d'effacement de plusieurs fichiers à la fois.
On aurait pu effacer tous les fichiers par l'ordre

ERASE *.*

Dans ce cas, il y aurait eu un message de confirmation du type.

ERASE *.* (Y/N)?

ERASE
CP/M Plus

Effacer des fichiers avec confirmation pour chacun.

Syntaxe

ERASE fich?*[C]

Ordre transitoire.
L'option C pour Confirmation lancera une question de sécurité avant chaque effacement.

Exemple

```
A>dir
A: LIST1          : LIST2          : PIP      COM : ERASE    COM
A>


A>era *.*[c]
A: LIST1    .     (Y/N)? y
A: LIST2    .     (Y/N)? y
A: PIP      .COM (Y/N)? n
A: ERASE    .COM (Y/N)? n


A>dir
A: PIP      COM : ERASE    COM
A>
```

Selon la réponse Y ou N, le fichier est conservé ou effacé, ce que confirme le directory de fin d'exemple.

Le truc

C'est cet ordre avec cette option que l'on utilisera pour "nettoyer" une disquette des fichiers parasites ou ayant servi au développement d'une application.

REN

Renommer un fichier.

Syntaxe

REN fich2=fich1

Ordre résident.

Cet ordre permet de changer le nom d'un fichier ; il ne peut fonctionner qu'avec un nom de fichier sans ambiguïté.

Exemple

```
A>dir
A: PIP      COM : ERASE    COM
A>

A>ren copy.com=pip.com


A>dir
A: COPY     COM : ERASE    COM
A>
```

RENAME

CP/M Plus

Renommer un fichier sur dialogue.

Syntaxe

RENAME

Ordre transitoire.

Cet ordre a le même effet que REN ; il est peut-être plus facile d'utilisation.

Exemple

```
A>dir
A: RENAME   COM : TXT1    TXT : TXT2    TXT : PIP    COM
A: TXT3     TXT : ERASE   COM


A>RENAME
Enter New Name: TEXTE1.TXT
Enter Old Name: TXT1.TXT


A>DIR
A: RENAME   COM : TEXTE1  TXT : TXT2    TXT : PIP    COM
A: TXT3     TXT : ERASE   COM
A>
```

On voit que l'effet est un simple changement de nom ; le dialogue lancé par l'ordre est cependant plus convivial.

RENAME

Renommer globalement certains fichiers.

Syntaxe

RENAME fich?*=fich?*

Ordre transitoire.

Cet ordre permet de changer le nom d'une série de fichiers en une autre série de noms, sous certaines conditions.

Exemple

```
A>DIR
A: RENAME   COM : FIC1    TXT : FIC2    TXT : PIP    COM
A: FIC3     TXT : ERASE   COM


A>RENAME TXT*.TXT=FIC*.TXT

TXT1    .TXT=FIC1    .TXT
TXT2    .TXT=FIC2    .TXT
TXT3    .TXT=FIC3    .TXT


A>dir
A: RENAME   COM : TXT1    TXT : TXT2    TXT : PIP    COM
A: TXT3     TXT : ERASE   COM
```

Le truc

Il est impératif que les noms d'origine et les noms de fin aient le même nombre de caractères avant le caractère * d'ambiguïté.

TRANSFERTS ENTRE PÉRIPHÉRIQUES

GET
CP/M Plus

Donner automatiquement des arguments à un ordre.

Syntaxe

GET FILE fich

Ordre transitoire.

Exemple

```
B>TYPE IN
TEXTEBIS
TEXTE3.TXT

B>DIR
B: C1OCPM3  EMS : GET       COM : SHOW      COM : TEXTE    TXT
B: TEXTE3   TXT : INITDIR   COM : RENAME    COM : IN
B: DIR      COM : DATE      COM : PIP       COM : SET      COM
SYSTEM FILE(S) EXIST
B>GET FILE IN
Getting console input from file: IN.
B>RENAME
Enter New Name: TEXTEBIS
Enter Old Name: TEXTE3.TXT

B>DIR
B: C1OCPM3  EMS : GET       COM : SHOW      COM : TEXTE    TXT
B: TEXTEBIS     : INITDIR   COM : RENAME    COM : IN
B: DIR      COM : DATE      COM : PIP       COM : SET      COM
SYSTEM FILE(S) EXIST

B>
```

Le fichier IN contient deux lignes de texte : le mot TEXTEBIS et le mot TEXTE3.TXT.

On vérifie qu'au directory figure le fichier de nom TEXTE3.TXT.

L'ordre GET FILE IN demande à CP/M de prendre le fichier IN comme réponse aux questions produites par la prochaine commande.

L'ordre RENAME demande deux arguments : le nouveau nom du fichier et l'ancien.

L'ordre s'exécute sans intervention ; les réponses aux questions sont données par le système.

Le dernier ordre DIR vérifie que le nouveau nom de fichier figure.

Le truc

Cette possibilité est à utiliser lorsque l'on a des configurations à effectuer, comme notamment un dialogue d'initialisation avec un serveur de base de données. Le fichier, en entrée, introduira les numéros de code, les divers identificateurs, de manière automatique et sans erreur.

Abréviation :

GET FI.

GET
CP/M Plus

Lancer une suite d'ordres CP/M.

Syntaxe

GET FILE fich [SYSTEM]

Ordre transitoire.
Possibilités voisines de l'ordre SUBMIT.

Exemple

```
B>TYPE LISTE
DATE SET
01/01/86
12:12:00
date
DIR

B>GET FILE LISTE [SYSTEM]
Getting console input from file: LISTE.
B>DATE SET

Enter today's date (MM/DD/YY): 01/01/85
Enter the time (HH:MM:SS):     12:00:00
Press any key to set time

B>date
Tue 01/01/85 12:00:01
B>dir
B: C1OCPM3  EMS : GET      COM : SHOW     COM : TEXTE    TXT
B: LISTE        : TEXTEBIS     : INITDIR  COM : RENAME   COM
B: IN           : DIR      COM : DATE     COM : PIP      COM
B: SET      COM

B>
```

Le fichier LISTE contient une suite d'ordres et d'arguments (l'ordre DATE SET suivi des arguments de date et d'heure).

Le fichier est exécuté, les ordres sont effectués, les arguments sont utilisés.

Cet ordre permet donc, en plus de l'ordre SUBMIT, de passer des arguments ; ici la date et l'heure nécessaires à l'ordre DATE SET.

PUT
CP/M Plus

Conserver une sortie écran sur disque.

Syntaxe

PUT CONSOLE FILE fich

Ordre transitoire.

Cet ordre stocke sur disque tous les caractères émis vers l'écran lors de l'exécution de l'ordre qui suit l'ordre PUT.

Le stockage s'arrête dès l'envoi d'un autre ordre au clavier.

Exemple

```
A>PUT CONSOLE FILE LISTE
Putting console output to file: LISTE.
A>DIR
A: C10CPM3   EMS : DIR       COM : RENAME    COM : IN
A: TEXTE1    TXT : SHOW      COM : TEXTE2    TXT : TEXTE3    TXT
A: PUT       COM : LISTE
A>
A>TYPE LISTE
A: C10CPM3   EMS : DIR       COM : RENAME    COM : IN
A: TEXTE1    TXT : SHOW      COM : TEXTE2    TXT : TEXTE3    TXT
A: PUT       COM : LISTE

A>
```

L'ordre PUT CONSOLE FILE LISTE envoie sur le fichier LISTE tout ce qui a été émis à l'écran.

L'ordre DIR provoque l'affichage du directory.

L'ordre TYPE LISTE permet de vérifier que PUT a créé un fichier qui est le témoin de tout ce qui est sorti à l'écran.

Remarque. – L'ordre PUT PRINTER FILE fich stocke sur le fichier disque fich tous les caractères émis vers l'imprimante.

Le truc

Cet ordre est très utile dans de nombreux cas où l'on veut conserver une trace d'un travail sans faire de sortie imprimante, notamment lors de dialogues télématiques. Il est toujours possible de reprendre le fichier témoin en traitement de texte sur l'imprimante... ou de l'effacer.

PIP

CP/M Plus

Dupliquer un fichier sur le même disque.

Syntaxe

PIP fich1=fich2

Ordre transitoire.

Cet ordre permet de créer un double d'un fichier sous un autre nom.

Il n'est évidemment pas possible de dupliquer un fichier sous son nom ; le système est en effet incapable de gérer deux noms de fichier identiques.

Exemple

```
A>DIR
A: C10CPM3   EMS : TEXTE1    TXT : PIP       COM
A>
A>PIP TEXTE2.TXT=TEXTE1.TXT


A>DIR
A: C10CPM3   EMS : TEXTE1    TXT : PIP       COM : TEXTE2    TXT
A>
```

On voit que le fichier TEXTE2.TXT a été créé ; il est le double de TEXTE1.TXT.

PIP

Copier un fichier d'un disque sur un autre avec un seul lecteur.

Syntaxe

PIP B:fich = A:fich

où A et B sont le nom de deux unités de disques ; fich est le nom du fichier à copier du disque A vers le B.

Ordre transitoire.

Exemple

```
A>DIR
A: C1OCPM3   EMS : PIP      COM : TEXTE2   TXT : TEXTE1   TXT

A>DIR B:
B: C1OCPM3   EMS : PIP      COM

A>PIP B:TEXTE1.TXT=A:TEXTE1.TXT


A>DIR
A: C1OCPM3   EMS : PIP      COM : TEXTE2   TXT : TEXTE1   TXT

A>DIR B:
B: C1OCPM3   EMS : PIP      COM : TEXTE1   TXT
A>
```

Les deux DIR donnent le contenu des disquettes A et B.

L'ordre PIP assure la copie entre les deux disques.

Les deux ordres DIR confirment que l'échange de fichiers s'est bien effectué entre les deux disques.

Le truc

La facilité de double nom offerte par l'Amstrad permet une grande souplesse pour cet ordre, puisque la copie est possible d'une disquette à une autre avec un seul lecteur, sous réserve d'obéir aux messages déroulants donnés par le système et invitant à changer les disquettes.

Avec cet ordinateur, prenez simplement la précaution de marquer les disques de la lettre dont ils sont affectés (A ou B).

PIP

Copier un fichier d'un disque à un autre en changeant son nom.

Syntaxe

PIP B:fich1=A:fich2

Ordre transitoire.

Il est possible de faire le transfert d'un fichier d'un disque à un autre en changeant son nom ; il faut se souvenir que le nom à gauche du signe = est le nom de la destination, et celui de droite le nom de la source.

Exemple

```
A>DIR
A: C1OCPM3  EMS : PIP      COM : TEXTE2   TXT : TEXTE1   TXT
A>DIR B:
B: C1OCPM3  EMS : PIP      COM

A>PIP B:TEXTE.BIS=A:TEXTE1.TXT

A>DIR
A: C1OCPM3  EMS : PIP      COM : TEXTE2   TXT : TEXTE1   TXT
A>DIR b:
B: C1OCPM3  EMS : PIP      COM : TEXTE    BIS
A>
```

On voit que le fichier TEXTE1.TXT du disque A a été copié sur B, sous le nom TEXTE.BIS.

Le truc

Lorsque l'on veut copier un fichier d'un disque à un autre sans changer son nom, il est possible de ne donner à gauche du signe (=) que le nom du lecteur destination : PIP B:=A:TEXTE1.TXT copie sur le disque B le fichier TEXTE1.TXT du disque A, sans changer son nom.

PIP

Regrouper plusieurs fichiers en un seul.

Syntaxe

PIP FICH=fich1,fich2,fich3...

Ordre transitoire.

Cette opération de concaténation permet de regrouper plusieurs fichiers que l'on aura, par exemple, développés séparément.

Exemple

```
A>DIR
A: C1OCPM3  EMS : PIP      COM : ES2          : ES1
A: ES3
A>type es1
Ceci est un texte
A>type es2
compose de
A>type es3
 trois parties.

A>pip estot=es1,es2,es3

A>type estot
Ceci est un texte compose de trois parties.
A>
```

L'exemple est assez explicite ; le premier directory montre trois fichiers ES1, 2 et 3, l'ordre TYPE permet de voir le contenu de chacun.

L'ordre PIP les regroupe dans l'ordre 1, 2 et 3 dans le fichier ESTOT.

L'ordre TYPE ESTOT permet de vérifier le regroupement des trois fichiers élémentaires.

Le truc

Cette possibilité de PIP est parfois utilisée par les développeurs (en BASIC par exemple). Chaque groupe de développement programme une partie en restant dans les limites définies (plage de numéros de lignes en BASIC, respect des noms de variables...), les programmes sauvés en format texte (option A du BASIC) sont – à l'aide de PIP – regroupés en un seul qui est le programme opérationnel.

PIP

Créer un fichier texte.

Syntaxe

PIP fich=CON:

Ordre transitoire.

Cet ordre place dans le fichier (fich) les caractères issus de la console, donc du clavier (CON:).

Exemple

```
A>DIR
A: C10CPM3  EMS : PIP      COM
A>PIP TEXTE=CON:
 On frappe un texte au clavier, termine par CTRL Z

A>DIR
A: C10CPM3  EMS : PIP      COM : TEXTE
A>TYPE TEXTE
 On frappe un texte au clavier, termine par CTRL Z
A>
```

Le texte frappé doit être terminé par le code CTRL Z, qui est le code de fin de fichier ; à ce moment-là, il est écrit sur disque.

Attention. – Aucune faute de frappe n'est permise ; elle sera écrite sur disque. Les corrections seront aussi mémorisées (CTRL H permet de reculer d'un caractère) et risquent d'apparaître en surimpression.

Le truc

Cette possibilité est intéressante par sa facilité de mise en œuvre pour des fichiers courts, comme les fichiers SUBMIT d'enchaînement d'ordres.

PIP

Copier un fichier disque sur une unité physique.

Syntaxe

PIP phys:=fich

Ordre transitoire.
Le contenu du fichier de nom fich est affiché sur le périphérique de nom phys:.

Exemple

```
A>PIP LST:=TEXTE
 On frappe un texte au clavier, termine par CTRL Z
A>
```

L'ordre a été frappé au clavier, l'affichage s'est fait uniquement à l'imprimante.
Si CON: avait été placé au lieu de LST:, l'affichage aurait été fait sur l'écran, effet identique à l'ordre TYPE.

Le truc

Cet ordre est utile, si vous possédez un interface série RS 232, pour envoyer un fichier vers un autre ordinateur, le périphérique s'appelant alors SIO:.

PIP

Dérouter les caractères émis par un périphérique vers un autre.

Syntaxe

PIP phys1:=phys2:

Ordre transitoire.
Les caractères émis par phys2: sont acheminés vers phys1:.

Exemple

```
A>PIP LST:=CON:
LLiiggnnee  11
LLiiggnnee  22
LLiiggnnee  33
A>
```

```
Ligne 1
Ligne 2
Ligne 3
```

On voit que les caractères émis par CON: (le clavier) s'affichent en double sur LST: (l'imprimante). Ceci est dû à la double sortie active (CTRL P) vers l'imprimante qui affiche un caractère, l'ordre PIP affichant le second.

Le second exemple montre l'affichage obtenu avec CTRL P inactif ; dans ce cas, l'ordre n'apparaît pas sur l'imprimante, mais les caractères frappés sur le clavier y sont affichés.

Le passage à la ligne se fait par CTRL M plus CTRL J, la fin de la sortie imprimante par CTRL Z.

PIP

Séquence de plusieurs transferts successifs.

Syntaxe

PIP

Ordre transitoire.

L'ordre PIP, sans argument, lance une succession de transferts ; les opérations s'arrêteront par un ordre vide.

Exemple

```
A>DIR
A: C10CPM3  EMS : PIP      COM : TEXTE
A>PIP
CP/M 3 PIP VERSION 3.0
*TEXTE1=texte
*TEXTE2=texte
*

A>DIR
A: C10CPM3  EMS : PIP      COM : TEXTE          : TEXTE1
A: TEXTE2
A>
```

L'ordre PIP lance d'abord un court message signalant que l'on se trouve sous l'utilitaire de transfert.

Chaque ligne de commande sera commencée par un astérisque, après lequel il suffira de frapper les arguments de l'ordre.

Au troisième astérisque, on frappe un retour-chariot simple pour signaler la fin de la procédure.

Le dernier ordre DIR sert ici à vérifier que les transferts se sont correctement effectués.

PIP
CP/M Plus

Copier une série de fichiers avec confirmation pour chacun.

Syntaxe

PIP fich1+fich2[C]

Ordre transitoire.

L'option C de l'ordre PIP demandera confirmation de copie pour chaque fichier.

Exemple

```
A>DIR
A: C10CPM3  EMS : TEXTE    TXT : PIP      COM : TEXTE2   TXT
A: TEXTE1   TXT
A>PIP B:=A:*.TXT[C]

COPYING -
TEXTE.TXT (Y/N)? y
TEXTE2.TXT (Y/N)? n
TEXTE1.TXT (Y/N)? y


A>dir b:
B: C10CPM3  EMS : PIP      COM : TEXTE    TXT : TEXTE1   TXT
A>
```

L'opération lancée, le système demande pour chaque fichier de confirmer la copie par Y pour oui, ou N pour non.

On vérifie par un directory que seuls les fichiers TEXTE.TXT et TEXTE1.TXT ont été copiés.

Le truc

Cette option est utilisée pour donner le résultat d'un travail à une personne, ou pour se constituer des disques sauvegarde d'un travail ; on fait l'ordre PIP B:=A:*.*[C], puis on sélectionne les fichiers que l'on sauvegarde en frappant Y à leur nom.

PIP

Copier des fichiers entre deux zones USER.

Syntaxe

PIP fich1[Ga]=fich2[Gb]

Ordre transitoire.

Fich1 est le fichier destination dans la zone USER numéro a, fich2 est le fichier source dans la zone USER b. Si l'un des noms de fichiers n'a pas de numéro de zone mentionné, la zone active du disque actif est prise par défaut.

En l'absence de spécification de zone, la zone 0 est prise par défaut.

Exemple

```
A>PIP TEXTE.TXT[G2]=TEXTE.TXT

A>USER 2
2A>DIR
A: TEXTE    TXT
2A>
2A>USER 0
A>
```

Sur le disque actif, le fichier Texte.txt de la zone 0 (par défaut) est recopié sur la zone 2, sous le même nom.

Le directory de la zone 2 donne confirmation du transfert.

PIP
CP/M Plus

Copier sans affichage à l'écran.

Syntaxe

PIP fich?*1=fich?*2[K]

Ordre transitoire.

Lorsque l'on a un transfert d'un grand nombre de fichiers à effectuer, on peut demander de ne pas afficher les noms des fichiers copiés.

Exemple

```
A>dir
A: C1OCPM3  EMS : TEXTE    TXT : PIP     COM : TEXTE2   TXT
A: TEXTE1   TXT
A>dir b:
B: C1OCPM3  EMS : PIP      COM
A>pip b:=*.TXT

COPYING -
TEXTE.TXT
TEXTE2.TXT
TEXTE1.TXT


A>pip b:=a:*.TXT[K]

A>dir b:
B: C1OCPM3  EMS : PIP      COM : TEXTE   TXT : TEXTE2   TXT
B: TEXTE1   TXT
A>
```

Dans l'exemple 1, on voit une opération normale de transfert générique des fichiers d'extension TXT: le système donne les noms des fichiers transférés.

Dans l'exemple 2, l'option K supprime la sortie à l'écran des noms de fichiers copiés ; cependant, l'ordre DIR confirme qu'ils ont bien été copiés sur le disque B.

PIP

Copier un texte transformé en majuscules.

Syntaxe

PIP fich1=fich2[U]

Ordre transitoire.

Certains fichiers, notamment des sources de programmes, gagnent à être écrits seulement en majuscules ; l'option U de PIP permet cette transformation.

Exemple

```
A>type texte.txt
Ceci est un texte d'essai.


A>PIP TEXTE.BIS=TEXTE.TXT[U]

A>
A>TYPE TEXTE.BIS
CECI EST UN TEXTE D'ESSAI.
A>
```

On voit que le fichier TEXTE.BIS copie de TEXTE.TXT, est en majuscules seulement.

PIP

Copier un fichier en numérotant les lignes.

Syntaxe

PIP fich1=fich2[N]

Ordre transitoire.

Dans le fichier recopié, chaque ligne sera précédée d'un numéro de ligne.

Exemple

```
A>TYPE TEXTE1.TXT
Ligne numero 1
Ligne numero 2
Ligne numero 3
Ligne numero 4

A>PIP TEXTE1.BIS=TEXTE1.TXT[N]


A>TYPE TEXTE1.BIS
     1: Ligne numero 1
     2: Ligne numero 2
     3: Ligne numero 3
     4: Ligne numero 4
A>
```

On voit que le fichier a été recopié dans TEXTE1.BIS en faisant précéder chaque ligne de son numéro.

L'option N2 au lieu de N permet une numérotation à six chiffres : 000001 au lieu de 1.

Le truc

Cette option est utilisée pour le travail à plusieurs sur un texte long : il est plus commode de dicter ses corrections en faisant référence à un numéro de ligne.

PIP

Copier une partie seulement d'un fichier.

Syntaxe

PIP fich1=fich2[Sch1^Z Qch2^Z]

Ordre transitoire.

Le fichier fich1 sera la copie de fich2, mais ne commencera qu'à la chaîne de caractères ch1 incluse et se terminera à la chaîne ch2 incluse.

Chaque chaîne (de début et de fin) doit être terminée par le caractère Contrôle Z.

Exemple

```
A>TYPE COPYR
CP/M Version 3.0COPYRIGHT 1982, DIGITAL RESEARCH

A>PIP COPYR.BIS=COPYR[SCOPY^Z Q82^Z]

A>TYPE COPYR.BIS
COPYRIGHT 1982

A>
```

On voit que le fichier COPYR.BIS est la copie de COPYR commençant à COPY et se terminant à 82.

Le truc

Attention, les commandes données sous CP/M Plus ne peuvent être qu'en majuscules ; si elles sont données en minuscules, elles sont obligatoirement traduites en majuscules. Il n'est donc pas toujours possible de travailler avec cet ordre de copie partielle sur des fichiers où les chaînes de début et de fin sont en minuscules.

PIP

Copier des fichiers avec vérification de la copie.

Syntaxe

PIP fich?*1=fich?*2[V]

Ordre transitoire.

Si les fichiers sont de grande importance, il est possible de vérifier la similitude de la copie avec l'original.

Exemple

```
A>dir
A: C1OCPM3  EMS : PIP      COM : TEXTE2   TXT : TEXTE    TXT
A: TEXTE1   TXT : COPYR
A>dir b:
B: C1OCPM3  EMS : PIP      COM


A>PIP b:=texte.txt[V]

A>dir b:
B: C1OCPM3  EMS : PIP      COM : TEXTE    TXT
A>
```

Le système fait une vérification bit par bit de la copie : s'il y a un problème, il le signale.

Le truc

La fiabilité des supports actuels rend cette précaution statistiquement moins indispensable.

DES INFORMATIONS SUR LE DISQUE

SHOW

CP/M Plus

Quelle est la place disponible sur les lecteurs ?

Syntaxe

SHOW

ou SHOW n:

Ordre transitoire.

Exemple

```
A>SHOW

A: RW, Space:        57k
B: RW, Space:        57k


A>show b:

B: RW, Space:       120k


A>
```

Le premier ordre, sans argument, donne les caractéristiques générales des deux lecteurs (qui contiennent les mêmes fichiers, d'où la similitude de taille disponible).

Le système donne l'état du disque (RW pour disque libre, RO si le disque était mis en lecture seulement).

La taille donnée est l'espace disponible en kilo-octets.

Le second ordre est spécifique à un lecteur particulier.

SHOW
CP/M Plus

Quel est le nom d'une disquette ?

Syntaxe

SHOW [LABEL]

Ordre transitoire.

Exemple

```
A>SHOW [LABEL]

Label for drive A:

Directory       Passwds  Stamp   Stamp
Label           Reqd     Create  Update  Label Created   Label Updated
------------    -------  ------  ------  --------------  --------------
CPM_PLUS.         off     off     off    12/15/82 00:39  10/15/85 11:07

A>
```

Cet ordre permet de visualiser le nom donné à la disquette par l'ordre SET ; il donne en outre des renseignements sur l'état de protection générale du disque, ainsi que la date de création du nom de la disquette et sa dernière modification.

SHOW
CP/M Plus

Quelles sont les zones USER utilisées.

Syntaxe

SHOW [USER]

Ordre transitoire.

Exemple

```
A>SHOW [USER]

A: Active User :  0
A: Active Files:  0   1   2
A: # of files  :  7   2   1

A: Number of free directory entries:      52


A>
```

La première ligne indique que la zone active est la zone 0 (zone par défaut, la première dans laquelle on se trouve lorsqu'on lance un disque).

La seconde ligne indique qu'il existe des fichiers dans les zones 0, 1 et 2.

La troisième ligne indique qu'il y a 7 fichiers dans la zone 0, 2 dans la zone 1 et 1 dans la zone 2.

La dernière ligne nous donne le nombre d'entrées directory encore disponibles (donc le nombre maximal de fichiers que l'on peut encore écrire sur la disquette).

SHOW

CP/M Plus

Quel est le nombre d'entrées directory disponibles ?

Syntaxe

SHOW [DIR]

Ordre transitoire.

Exemple

```
A>SHOW [DIR]

A: Number of free directory entries:          52

A>
```

CP/M gère le directory d'un certain nombre de fichiers. Le nombre de fichiers est un nombre fini, fonction de la taille de la disquette et de la taille des blocs élémentaires de transfert (en général 128 octets).

Dans notre cas, CP/M nous répond qu'il dispose encore de 52 entrées directory, donc qu'il est capable de créer et de gérer encore 52 fichiers.

SHOW

CP/M Plus

Quelles sont les caractéristiques d'une disquette ?

Syntaxe

SHOW [DRIVE]

ou

SHOW n: [DRIVE]

Ordre transitoire.

Exemple

```
A>SHOW [DRIVE]

        A: Drive Characteristics
    1,368: 128 Byte Record Capacity
      171: Kilobyte Drive  Capacity
       64: 32 Byte  Directory Entries
       64: Checked  Directory Entries
      128: Records / Directory Entry
        8: Records / Block
       36: Sectors / Track
        2: Reserved  Tracks
      512: Bytes / Physical Record


A>

A>SHOW b:[DRIVE]

        B: Drive Characteristics
    1,368: 128 Byte Record Capacity
      171: Kilobyte Drive  Capacity
       64: 32 Byte  Directory Entries
       64: Checked  Directory Entries
      128: Records / Directory Entry
        8: Records / Block
       36: Sectors / Track
        2: Reserved  Tracks
      512: Bytes / Physical Record


A>
```

Le premier exemple nous apprend :

Ligne par ligne :

1: Le nom du drive.
2: Le nombre de blocs élémentaires que contient la disquette (1386), et leur taille unitaire (128 octets).
3: La capacité totale du disque (171 kilo-octets).
4: Le nombre d'entrées directory possibles (64).
5: Le nombre d'entrées directory effectivement actives.
6: La taille d'un bloc d'informations.
7: Le nombre de blocs d'informations par secteurs de la disquette (8).
8: Le nombre de secteurs par piste (36).

9: Le nombre de pistes réservées pour le système (2).

10: Le nombre d'octets dans un enregistrement physique (un fichier ne peut pas faire moins).

Le second exemple montre que l'on peut obtenir les renseignements d'un disque précis en le nommant avant l'option.

Le truc

Les renseignements donnés par l'ordre SHOW [DRIVE] sont des renseignements physiques liés au support disquette précis de votre système et à son utilisation par CP/M Plus. On ne l'utilise donc que rarement, pour retrouver, se remémorer ces caractéristiques.

STAT
CP/M2

Obtenir la place disponible d'un disque.

Syntaxe

STAT
STAT␣D:

Exemple

```
A>STAT

A: RW, Free Space:          164k

A>
```

L'ordre STAT simple donne pour le disque actif :
- sa situation en écriture/lecture (R/W) ou en lecture seule (R/O) ;
- la place en kilo-octets restant disponible.

Exemple

```
A>B:
B>STAT

A: RW, Free Space:          164k
B: RW, Free Space:          164k

B>
```

Si le disque A est actif et si l'on rend le disque B actif à son tour, l'ordre simple STAT donne les caractéristiques des deux disques.

Un ordre STAT sous A actif donnera le même résultat si le B a été rendu actif antérieurement (ordre DIR B: par exemple).

Exemple

```
A>STAT A:

A: RW, Free Space:        164k
A>
```

L'ordre STAT suivi de l'indicatif du lecteur (ici A:) donne une réponse différente : seule la place disponible apparaît.

STAT
CP/M2

Obtenir les caractéristiques physiques d'un disque.

Syntaxe

STAT␣D:DSK:

Exemple

```
A>STAT A:DSK:

        A: Drive Characteristics
    2,528: 128 Byte Record Capacity
      316: Kilobyte Drive  Capacity
       64: 32 Byte  Directory Entries
       64: Checked  Directory Entries
      256: 128 Byte Records / Directory Entry
       16: 128 Byte Records / Block
       32: 128 Byte Records / Track
        1: Reserved  Tracks

A>
```

- ☐ **A** : est le nom du disque analysé.
- ☐ **1904** : est le nombre d'unités élémentaires de 128 octets disponibles sur le disque (ce sont les Recs de la réponse à l'ordre STAT nom de fichier vu plus loin.
- ☐ **238** : c'est le nombre de kilo-octets accessibles sur le disque (disponibles ou non).
- ☐ **64** : c'est le nombre d'entrées directory, existantes et vérifiées ; c'est donc aussi le nombre maximal de fichiers que l'on pourra stocker sur le disque.
- ☐ **128** : est le nombre d'enregistrements adressables par une seule entrée directory.
- ☐ **8** : nombre d'enregistrements par secteurs, ici 8 × 128 soit 1024 octets par secteur.
- ☐ **56** : est le nombre de secteurs par piste, on a donc 57344 octets par piste soit 56 kilo-octets (un kilo-octet vaut 1024 octets).
- ☐ **1** : est le nombre de pistes réservées pour le système.

STAT
CP/M2

Obtenir les caractéristiques physiques d'un fichier

Syntaxe

STAT␣FICHIER.EXT

Exemple

```
A>STAT ED.CMD

 Drive A:                                User :  0
 Recs   Bytes FCBs Attributes            Name
   72    10k     1 Sys RO              A:ED       .CMD
------------------------------------------------------.
Total:    10k     1
A: RW, Free Space:          154k
```

Pour le fichier MBASIC.COM, on obtient les informations suivantes :

- ☐ **Recs : 194** Le fichier occupe 194 secteurs élémentaires comportant chacun 128 octets.
- ☐ **Bytes : 25K** Le fichier occupe 25 kilo-octets.
- ☐ **Ext : 2** Il utilise 2 blocs de gestion de 25 kilo-octets, ce qui ne veut pas dire qu'il utilise 50 kilo-octets sur le disque, ces blocs n'occupant qu'une place fictive dans le système d'exploitation.
- ☐ **Acc : R/W** Le fichier est en lecture/écriture.

Le système, après avoir rappelé le nom du fichier A:MBASIC.COM, indique pour mémoire la place disponible sur le disque.

Si le fichier avait été rendu opaque au directory (ordre STAT $SYS), son nom apparaîtrait entre parenthèses.

STAT
CP/M2

Obtenir les caractéristiques physiques de tous les fichiers d'un disque.

Syntaxe

STAT␣D:*.*

Exemple

```
A>STAT *.*

 Drive A:                              User :  0
 Recs   Bytes FCBs Attributes           Name
  205    26k     1 Dir RW             A:ASM86   .CMD
   16     2k     1 Dir RW             A:ASSIGN  .CMD
   19     4k     1 Dir RW             A:COPYDISK.CMD
  109    14k     1 Dir RW             A:DDT86   .CMD
   72    10k     1 Sys RO             A:ED      .CMD
   14     2k     1 Dir RW             A:FUNCTION.CMD
   45     6k     1 Dir RW             A:GENCMD  .CMD
   52     8k     1 Dir RW             A:HELP    .CMD
  195    26k     1 Dir RW             A:HELP    .HLP
   50     8k     1 Dir RW             A:NEWDISK .CMD
   59     8k     1 Dir RW             A:PIP     .CMD
   14     2k     1 Dir RW             A:PROTOCOL.CMD
   14     2k     1 Dir RW             A:SPEED   .CMD
   73    10k     1 Dir RW             A:STAT    .CMD
   31     4k     1 Dir RW             A:SUBMIT  .CMD
   21     4k     1 Dir RW             A:TOD     .CMD
-----------------------------------------------------
Total:   136k   16
A: RW, Free Space:          154k
```

Tous les fichiers du disque ont été analysés, c'est le document-type que l'on classe dans la pochette du disque ; il permet à tout moment, sans avoir à charger le disque, de connaître en détail son contenu.

STAT
CP/M2

Obtenir les possibilités de l'ordre.

Syntaxe

STAT␣VAL:

Exemple

1re ligne : Possibilité d'écriture sur le disque sous la forme d:=R/O où d est la lettre du lecteur (A ou B en général).

2e ligne : Possibilités d'affectation de paramètres aux fichiers : le nom du fichier suivi des quatre paramètres possibles : $R/O, $R/W, $SYS, $DIR.

3ᵉ ligne : Syntaxes possibles pour obtenir l'état du disque actif ou d'un autre disque.
4ᵉ ligne : Syntaxe pour l'obtention des zones utilisateurs.
Lignes suivantes : Elles donnent les syntaxes possibles pour l'affectation de périphériques sur les quatre sorties logiques.

L'ordre STAT VAL : est donc un aide-mémoire des syntaxes de l'ordre STAT en général.

```
A>STAT VAL:
STAT 2.2

Read Only Disk: d:=RO
Set Attribute: d:filename.typ [ro] [rw] [sys] or [dir]
Disk Status  : DSK: d:DSK:
User Status  : USR: d:USR:
Iobyte Assign:
CON: = TTY: CRT: BAT: UC1:
AXI: = TTY: PTR: UR1: UR2:
AXO: = TTY: PTP: UP1: UP2:
LST: = TTY: CRT: LPT: UL1:

A>
```

STAT
CP/M2

Quelles sont les zones utilisateurs actives ?

Syntaxe

STAT␣USR:

Exemple

```
A>STAT USR:

A: Active User :  0
A: Active Files:  0  5 15

A>
```

Comme nous le verrons dans l'ordre USER de CP/M, le disque peut avoir plusieurs zones utilisateurs dans lesquelles l'opérateur peut aller travailler.

Ici, nous voyons (première ligne) que la zone utilisateur actuellement active est la zone zéro (celle qui est automatiquement active par défaut à la mise sous tension).

Nous voyons aussi que trois zones possèdent des fichiers (donc ont été effectivement utilisées), il s'agit des zones 0, 5 et 9.

STAT
CP/M2

Connaître les caractéristiques physiques des périphériques

Syntaxe

STAT␣DEV:

Exemple

```
A>STAT DEV:
CON: is TTY:
AXI: is TTY:
AXO: is TTY:
LST: is TTY:
```

Les quatre périphériques ont l'affectation standard que l'on retrouve à la mise sous tension de la plupart des machines.

Pour les quatre sorties logiques, les codes sont les suivants :

CON	: pour console (terminal écran/clavier).
RDR	: pour lecteur de rubans perforés.
ou **AXI**	: pour entrée auxiliaire.
PUN	: pour perforateur de rubans.
ou **AXO**	: pour sortie auxiliaire.
LST	: sortie de listes (imprimante).

Ces termes font partie de l'histoire de l'informatique, il faut les considérer ici comme des aide-mémoire caractérisant les quatre sorties.

Pour les périphériques, les codes sont les suivants :

TTY	: télétype, faible vitesse.
CTR	: terminal écran/clavier.
BAT	: lecteur par bloc (rubans ou cartes).
UC1	: console définie par l'utilisateur.
UR1 ou **UR2**	: lecteur à définir par l'utilisateur.
PTR ou **PTP**	: lecteur ou perforateur de bandes.
UP1 ou **UP2**	: perforateurs à définir par l'utilisateur.
LPT	: imprimante ligne.
UL1	: imprimante à définir par l'utilisateur.

STAT
CP/M2

Protéger un disque contre la destruction.

Syntaxe

STAT␣D:=R/O

Exemple

```
(1) A>STAT A:=R/O

    Drive A: set to Read Only (RO)

    A>STAT

    A: RO, Free Space:        156k
    B: RW, Free Space:        164k
```

(1) L'ordre protège le disque A contre l'écriture ou l'effacement.
Un ordre STAT simple montre que le disque est effectivement protégé (R/O).

Exemple

```
(2) A>^C
    A>STAT

    A: RW, Free Space:        156k
```

(2) Un départ à chaud (Contrôle C) remet le disque en position normale, la protection ne se situait donc pas sur le disque mais en mémoire vive de l'ordinateur.

Cette protection n'est pas définitive !

Exemple

```
(3) A>STAT A:=R/O

    Drive A: set to Read Only (RO)
```

On reprotège le disque.

Exemple

```
(4) A>ERA *.*
    ALL (Y/N)?Y
    Bdos Err On A: R/O

(5) A>STAT

    A: RW, Free Space:        156k

    A>
```

(4) Une tentative de destruction d'un ou de tous les fichiers (comme ici) est refusée par la machine.

Toute tentative de ce genre replace le disque en autorisation d'écriture comme le montre le (5).

Attention. – Pour protéger définitivement un disque, utilisez la possibilité donnée par la fenêtre de verrouillage du disque.

STAT
CP/M2

Rendre un fichier opaque au directory.

Syntaxe

STAT␣FICHIER.EXT␣$SYS

Exemple

(1)

```
A>DIR
A: ASM86     CMD : TOD       CMD : COPYDISK CMD : DDT86    CMD
A: ASSIGN    CMD : NEWDISK   CMD : FUNCTION CMD : PROTOCOL CMD
A: ED        CMD : SPEED     CMD : PIP      CMD : SUBMIT   CMD
A: HELP      CMD : HELP      HLP : STAT     CMD : GENCMD   CMD
```

(2)

```
A>STAT ED.CMD $SYS

A:ED      .CMD set to System (Sys)
A>
```

(3)

```
A>DIR
A: ASM86     CMD : TOD       CMD : COPYDISK CMD : DDT86    CMD
A: ASSIGN    CMD : NEWDISK   CMD : FUNCTION CMD : PROTOCOL CMD
A: SPEED     CMD : PIP       CMD : SUBMIT   CMD : HELP     CMD
A: HELP      HLP : STAT      CMD : GENCMD   CMD

SYSTEM FILE(S) EXIST
A>
```

(1) Un directory sur le disque A donne la liste des fichiers existants.
(2) On passe le fichier A:MBASIC.COM en invisible.
(3) Un nouveau directory ne le laisse pas apparaître.

Cette procédure est utilisée pour mettre des fichiers en sécurité ; nous verrons que cette sécurité n'est pas absolue, notamment sous un langage évolué.

STAT
CP/M2

Rendre un fichier accessible au directory.

Syntaxe

STAT␣FICHIER.EXT␣$DIR

Exemple

(1)

```
A>
A>DIR
A: ASM86     CMD : TOD       CMD : COPYDISK CMD : DDT86    CMD
A: ASSIGN    CMD : NEWDISK   CMD : FUNCTION CMD : PROTOCOL CMD
A: SPEED     CMD : PIP       CMD : SUBMIT   CMD : HELP     CMD
A: HELP      HLP : STAT      CMD : GENCMD   CMD

SYSTEM FILE(S) EXIST
```

(2)

```
A>STAT ED.CMD $DIR

A:ED      .CMD set to Directory (Dir)
```

(3)

```
A>DIR
A: ASM86    CMD : TOD      CMD : COPYDISK CMD : DDT86    CMD
A: ASSIGN   CMD : NEWDISK  CMD : FUNCTION CMD : PROTOCOL CMD
A: ED       CMD : SPEED    CMD : PIP      CMD : SUBMIT   CMD
A: HELP     CMD : HELP     HLP : STAT     CMD : GENCMD   CMD
A>
```

(1) Un directory du disque A donne les fichiers accessibles.
(2) On place le fichier en visible, la réponse :

MBASIC.COM set to DIR

montre que le fichier existait, sinon on aurait obtenu la réponse :

No file.

(3) Un nouveau directory laisse apparaître le fichier.

STAT
CP/M2

Protéger un fichier contre la destruction.

Syntaxe

STA␣FICHIER.EXT␣$R/0

Exemple

```
(1) A>

    STAT ED.CMD $R/O

    A:ED      .CMD set to Read Only (RO)
(2) A>ERA ED.CMD
    Bdos Err On A: File R/O
(3) A>ERA ED.CMD
    Bdos Err On A: File R/O
(4) A>^C
    A>ERA ED.CMD
    Bdos Err On A: File R/O
    A>
```

(1) On place le fichier XSUB.COM en lecture seulement, il ne pourra être ni effacé ni détruit.

(2) Une tentative d'effacement ne peut aboutir.

(3) Une nouvelle tentative n'aboutit toujours pas, donc contrairement à la protection du disque entier (STA A:=R/O), il n'y a pas de retour automatique en écriture.

(4) Un démarrage à chaud ne déverrouille pas davantage le fichier.

Contrairement à l'ordre de protection du disque entier qui était mémorisé en mémoire vive, l'ordre de protection d'un fichier est mémorisé sur le disque, il est définitivement acquis sauf ordre de déverrouillage explicite.

STAT

CP/M2

Déverrouiller un fichier protégé en écriture

Syntaxe

STAT␣FICHIER.EXT␣$R/W

Exemple

(1)

```
A>ERA ED.CMD
Bdos Err On A: File R/O
```

(2)

```
A>STAT ED.CMD $R/W

A:ED      .CMD set to Read Write (RW)
```

(3)

```
A>DIR
A: ASM86    CMD : TOD      CMD : COPYDISK CMD : DDT86    CMD
A: ASSIGN   CMD : NEWDISK  CMD : FUNCTION CMD : PROTOCOL CMD
A: ED       CMD : SPEED    CMD : PIP      CMD : SUBMIT   CMD
A: HELP     CMD : HELP     HLP : STAT     CMD : GENCMD   CMD
```

(4)

```
A>ERA ED.CMD
```

(5)

```
A>dir
A: ASM86    CMD : TOD      CMD : COPYDISK CMD : DDT86    CMD
A: ASSIGN   CMD : NEWDISK  CMD : FUNCTION CMD : PROTOCOL CMD
A: SPEED    CMD : PIP      CMD : SUBMIT   CMD : HELP     CMD
A: HELP     HLP : STAT     CMD : GENCMD   CMD
```

(1) Une tentative de destruction de XSUB.COM n'aboutit pas, le fichier est protégé.
(2) On déverrouille le fichier.
(3) On vérifie l'existence du fichier au directory.
(4) On tente de détruire le fichier, aucun message d'erreur n'apparaît.
(5) Le directory ne fait pas apparaître le fichier XSUB.COM qui a été détruit.

STAT
CP/M2

Cas particuliers des fichiers opaques et protégés – le BASIC.

Exemple

(1)

```
A>
A>STAT A:MBASIC.COM $SYS

MBASIC.COM SET TO R/O
```

(2)

```
A>STAT A:XSUB.COM $R/O

XSUB.COM SET TO R/O
```

(3)

```
A>MBASIC
BASIC-80 REV. 5.21

COPYRIGHT 1977-1981 (C) BY MICROSOFT
30455 BYTES FREE
OK
```

(4)

```
FILES
DUMP    .ASM  MBASIC  .COM  PIP    .COM XSUB  .COM
ED      .COM  ASM     .COM  DDT    .COM STAT  .COM
DUMP    .COM  COPYDISK.COM  FORMAT .COM
OK
```

Les fichiers opaques ou protégés apparaissent à l'ordre FILES du BASIC sous une syntaxe particulière.

Les fichiers opaques au directory sont visibles sous BASIC mais la seconde lettre de l'extension est en inversion vidéo ; ici, c'est le cas de MBASIC.COM qui a été caché en (1) et qui apparaît en (4).

Les fichiers protégés en écriture sont visibles à l'ordre FILES du BASIC mais la première lettre de leur extension est en inversion vidéo.

STAT
CP/M2

Modifier les caractéristiques physiques d'un périphérique.

Syntaxe

STAT␣CON:=CRT:

où CON est un exemple de nom d'entrée/sortie et CRT un exemple de caractéristique.

Exemple

```
(1) A>STAT DEV:
    CON: is TTY:
    AXI: is TTY:
    AXO: is TTY:
    LST: is TTY:

    A>
(2) A>STAT RDR:=CRT:

    Invalid Assignment

    A>
(3) A>STAT CON:=CRT:

(4) A>STAT DEV:
    CON: is CRT:
    AXI: is TTY:
    AXO: is TTY:
    _ST: is TTY:

    A>
```

(1) On demande les caractéristiques actuelles pour les quatre périphériques.

(2) On veut assigner CRT (Grande vitesse) à RDR, la machine le refuse (voir STAT VAL:), cette caractéristique n'est pas possible.

(3) On assigne CRT à CON (clavier/écran), la machine l'accepte.

(4) L'ordre STAT DEV: vérifie la modification de caractéristiques de CON.

Remarque. – Reportez-vous à l'ordre STAT DEV: pour la description et la signification des codes ; il faut remarquer que le plus souvent, les valeurs par défaut assignées par le constructeur donnent les meilleurs résultats en fonction des périphériques standards de la marque.

MANIPULATIONS DES ATTRIBUTS DE FICHIERS

SET

CP/M PLUS

Nommer un disque.

Syntaxe

SET [NAME=xxx]

Ordre transitoire.

Cet ordre permet d'affecter un nom (une étiquette) au disque ; ce nom est générique au support et lui sert en quelque sorte de titre.

Exemple

```
A>SET [NAME=CP/M]

ERROR: Unrecognized option.

Option: M
Label for drive A:

Directory          Passwds  Stamp    Stamp    Stamp
Label              Reqd     Create   Access   Update
--------------     -------  -------  -------  -------
A:CP          .      off      off      off      off


A>SET [NAME=CP_M]

Label for drive A:

Directory          Passwds  Stamp    Stamp    Stamp
Label              Reqd     Create   Access   Update
--------------     -------  -------  -------  -------
A:CP_M        .      off      off      off      off


A>
```

Le premier exemple illustre les problèmes venant de l'utilisation de caractères non alphanumériques dans un nom qui doit être géré par CP/M (nom de fichier ou, ici, mot de passe).

L'utilisation du trait de fraction dans le nom CP/M a dérouté CP/M qui considère le trait de fraction comme le signe d'une option de l'ordre. Ne reconnaissant pas l'option M, il le signale par un message et place comme nom les caractères placés avant (ici CP).

Le tableau suivant délivre l'état du titre de la disquette, le nom (CP), l'état de protection (pas de mot de passe requis), l'état des dateurs de fichiers, tous inactifs.

Le second exemple utilise un des rares signes acceptés dans les noms : le souligné ; tout marche correctement.

SET
CP/M PLUS

Donner un mot de passe à un ou des fichiers.

Syntaxe

SET fich?* [PASSWORD=xxx]

Ordre transitoire.

Exemple

```
A>SET TEXTE.TXT [PASSWORD=ABC PROTECT=READ]

A:TEXTE   .TXT  Protection = READ, Password = ABC

A>


A>type texte.txt


Password: ABC
Ceci est un texte d'essai.
A>


A>type texte.txt


Password: XXX

CP/M Error On A: Password Error
BDOS Function = 15  File = TEXTE   .TXT
A>
```

On voit que l'ordre SET renvoie les éléments de protection attribués au fichier.

Une tentative de lecture du fichier entraîne la demande du mot de passe qui ne s'affiche pas à l'écran durant la frappe. S'il est correct, il y a exécution de l'ordre, s'il est incorrect, il y a affichage d'un message d'erreur.

Le truc

Souvenez-vous bien de vos mots de passe !

Attention. – L'ordre de protection avec mot de passe doit être précédé par une mise en protection du disque à l'aide de l'ordre SET [PROTECT=ON]. Sinon, le système répond que le disque ne peut pas recevoir de protection pour ses fichiers.

Le mot de passe doit précéder l'option de protection.

Si on ne donne pas de mot de passe pour modifier une protection, le système le demandera.

SET

CP/M PLUS

Activer (ou désactiver) la protection d'un disque.

Syntaxe

SET [PROTECT=ON]

ou SET [PROTECT=OFF]

Ordre transitoire.

Exemple

```
A>SET [PROTECT=ON]

Label for drive A:

Directory       Passwds  Stamp    Stamp    Stamp
Label           Reqd     Create   Access   Update
--------------  -------  -------  -------  -------
A:CP_M     .      on       off      off      off


A>TYPE TEXTE.TXT


Password:
Ceci est un texte d'essai.
A>

A>SET [PROTECT=OFF]

Label for drive A:

Directory       Passwds  Stamp    Stamp    Stamp
Label           Reqd     Create   Access   Update
--------------  -------  -------  -------  -------
A:CP_M     .     off      off      off      off


A>type TEXTE.TXT
Ceci est un texte d'essai.
A>
```

L'ordre SET [PROTECT=ON] active la protection du disque actif (on aurait pu dire SET B:[PROTECT=ON]) ; une tentative de lister un fichier de degré de protection lecture (READ) entraîne la demande de mot de passe. Ce mot de passe n'apparaît pas à l'écran lors de sa frappe.

L'ordre SET [PROTECT=OFF] désactive globalement la protection du disque.

Le truc

L'ordre de protection/déprotection ne sert à rien si vous n'avez pas lancé l'ordre SET [PASSWORD=xxx] au préalable : une protection du disque sans mot de passe suppose que n'importe qui puisse également déprotéger sans mot de passe.

Si le disque est affecté d'un mot de passe, celui-ci sera réclamé avant toute opération de déprotection.

Attention. – Si la protection en lecture affecte un fichier exécutable (extension .COM), il ne pourra être lancé que sur son nom complet (comportant son mot de passe).

Exemple : le fichier DATE.COM est protégé par le mot de passe ABC ; pour utiliser le fichier, il faut taper

DATE ;ABC.

SET
CP/M

Annuler le mot de passe d'un fichier.

Syntaxe

SET fich [PASSWORD=

Ordre transitoire.

Exemple

```
A>SET TEXTE.TXT [PA=


A:TEXTE   .TXT
Password ?

A:TEXTE   .TXT  Protection = NONE, Password =
```

Pour annuler un mot de passe, il suffit d'entrer un simple retour-chariot après le signe égal du mot "PASSWORD".

Pour replacer le mot de passe à blanc, le système demande évidemment l'ancien mot de passe.

On voit que le fait d'annuler le mot de passe annule aussi la hiérarchie de protection qui repasse à NONE (pas de protection).

Cet ordre est également aussi utilisable pour annuler le mot de passe d'un disque ; dans ce cas, on écrit :

SET [PASSWORD=

Le truc

Dans notre exemple, nous avons utilisé l'abréviation PA pour PASSWORD.

SET

Modifier le niveau de protection des fichiers

Syntaxe

SET fich[PROTECTION=nn]

où nn=READ, WRITE ou DELETE.

Ordre transitoire.

Exemple

```
A>SET TEXTE.TXT [PA=CBA

A:TEXTE   .TXT  Protection = READ, Password = CBA
```

```
A>SET TEXTE.TXT [PR=WRITE]


A:TEXTE    .TXT
Password ?

A:TEXTE    .TXT  Protection = WRITE

A>
A>ERASE TEXTE.TXT
A: TEXTE    .TXT  Not erased,  Password Error
Password:
A: TEXTE    .TXT  Not erased,  Password Error

A>
```

Placer un mot de passe, sans option, place le fichier en protection maximale (READ).

On peut modifier le niveau de protection par l'option PROTECT=.

L'option PROTECT= n'est active que si le fichier possède un mot de passe.

On voit qu'une tentative d'effacement d'un fichier protégé en écriture est inutile si l'on ne dispose pas du mot de passe.

Le truc

L'exemple utilise l'abréviation PA pour password et PR pour protect.

SET

CP/M Plus

Placer (ou libérer) un disque en lecture seule.

Syntaxe

SET [RO]

ou SET [RW]

Ordre transitoire

Exemple

```
A>SET [RO]

Drive A: set to Read Only (RO)

A>ERASE PIP.COM

Drive Read Only
```

```
A>ERASE PIP.COM

Drive Read Only
A>SET [RW]

Drive A: set to Read Write (RW)

A>ERASE ESS.TXT
A>
```

La protection du disque est valable pour tous les fichiers ; aucun ne peut être effacé.

Cette protection est de hiérarchie supérieure à la protection des fichiers par mot de passe : une tentative d'effacement d'un fichier protégé sur un disque protégé générera le message de disque protégé.

La protection prend fin par l'ordre SET [RW], ou lors de la réinitialisation du disque (la protection n'est pas inscrite sur le disque).

SET

Modifier les attributs directory d'un fichier.

Syntaxe

SET fich?* [SYS]

ou SET fich?* [DIR]

Ordre transitoire.

Exemple

```
A>SET *.COM [SYS]

A:PIP     .COM  set to system (SYS), Read Write (RW)
A:SHOW    .COM  set to system (SYS), Read Write (RW)
A:SET     .COM  set to system (SYS), Read Write (RW)
A:DATE    .COM  set to system (SYS), Read Write (RW)
A:TYPE    .COM  set to system (SYS), Read Write (RW)
A:ERASE   .COM  set to system (SYS), Read Write (RW)

A>dir
A: C10CPM3  EMS : TEXTE    TXT
SYSTEM FILE(S) EXIST
A>DIRS
A: PIP      COM : SHOW     COM : SET      COM : DATE     COM
A: TYPE     COM : ERASE    COM
NON-SYSTEM FILE(S) EXIST
A>
```

L'ordre de mise en système (SYS) peut porter sur un seul fichier, ou sur une série, comme c'est le cas ici. Les fichiers sont listés, un à un, avec les états qui leur sont attribués.

Pour repasser des fichiers en directory (DIR), l'argument de l'option devient DIR au lieu de SYS.

Le truc

L'attribut SYS place les fichiers en ressource commune de toutes les zones USER.

SET

CP/M Plus

Activer (ou désactiver) la fonction ARCHIVE.

Syntaxe

SET fich?* [ARCHIVE=ON]

ou SET fich?* [ARCHIVE=OFF]

Ordre transitoire.

La fonction ARCHIVE est une facilité de CP/M Plus permettant de copier seulement des fichiers qui ont été modifiés (sauvegarde de sécurité). L'utilisation de cette fonction est reprise dans le chapitre Tours de main.

Exemple

```
A>
SET *.TXT [ARCHIVE=ON]


A:TEXTE2  .TXT
Password ?

A:TEXTE2  .TXT  set to directory (DIR), Read Write (RW) A
A:TEXTE   .TXT  set to directory (DIR), Read Write (RW) A
A:TEXTE1  .TXT  set to directory (DIR), Read Write (RW) A

A>DIR [FULL]

Scanning Directory...

Sorting  Directory...

Directory For Drive A:  User  0

    Name      Bytes  Recs  Attributes  Prot      Update          Access
------------ ------ ------ ------------ ------ --------------  --------------

C1OCPM3  EMS    25k    200 Dir RW       None
DIR      COM    15k    114 Dir RW       None                    12/15/82 00:21
ERASE    COM     4k     29 Dir RW       None                    12/15/82 00:00
INITDIR  COM    32k    250 Dir RW       None
PIP      COM     9k     68 Dir RW       None
SET      COM    11k     81 Dir RW       None                    12/15/82 00:21
SETDEF   COM     4k     32 Dir RW       None
SHOW     COM     9k     66 Dir RW       None
TEXTE    TXT     1k      1 Dir RW Arcv  Read
TEXTE1   TXT     1k      1 Dir RW Arcv  Read
TEXTE2   TXT     1k      2 Dir RW Arcv  Read
TYPE     COM     3k     24 Dir RW       None

Total Bytes     =    115k  Total Records =     868  Files Found =   12
Total 1k Blocks =    115   Used/Max Dir Entries For Drive A:   27/  64

A>
```

On voit qu'un fichier protégé par mot de passe ne peut être modifié au niveau du directory que sur fourniture de ce mot de passe.

La modification d'attribut se fait avec annonce à l'écran. La lettre A en fin de ligne signale le passage en attribut ARCHIVE.

L'ordre DIR [FULL] montre les trois fichiers avec l'attribut Arcv (ARCHIVE).

Le truc

L'attribut ARCHIVE est à rapprocher de l'ordre PIP avec option [A].

SET

Donner un niveau de datation aux fichiers.

Syntaxe

SET [Option=ON]

Ordre non résident.

Les options peuvent être CREATE, UPDATE ou ACCESS ; l'argument peut être ON ou OFF.

Exemple

```
A>SET [CREATE=ON]


Directory Label
Password ?

Label for drive A:

Directory        Passwds  Stamp    Stamp    Stamp
Label            Reqd     Create   Access   Update
--------------   -------  -------  -------  -------
A:LABEL    .       on       on       off      off


A>PIP ESS=CON:
jhgjhgf
A>dir [FULL]

Scanning Directory...

Sorting  Directory...

Directory For Drive A:  User  0

    Name      Bytes   Recs   Attributes   Prot      Update          Create
------------ ------ ------ ------------ ------ --------------  --------------

C1OCPM3  EMS    25k    200 Dir RW Arcv  None                    12/15/82 00:53
DIR      COM    15k    114 Dir RW       None                    12/15/82 00:03
ED       COM    10k     73 Dir RW Arcv  None                    12/15/82 00:57
ESS              1k      1 Dir RW       None                    10/18/85 12:12
PIP      COM     9k     68 Dir RW Arcv  None                    12/15/82 00:48
SET      COM    11k     81 Dir RW Arcv  None                    12/15/82 00:04
TEXTE2   BAK     1k      1 Dir RW Arcv  None                    11/17/85 21:29
TEXTE3           1k      1 Dir RW Arcv  None                    12/15/82 00:09

Total Bytes     =     73k  Total Records =     539  Files Found =    8
Total 1k Blocks =     73   Used/Max Dir Entries For Drive A:   15/  64

A>
```

La modification d'un directory protégé entraîne la demande de mot de passe.

L'ordre SET renvoie l'état des options ; ici, on voit que le mot de passe et le dateur de création sont actifs.

L'ordre DIR [FULL] donne les dates de création ; on voit que notre fichier ESS, créé sous PIP, est daté en création à la date et à l'heure du jour.

Le truc

Les options ACCESS et CREATE sont exclusives l'une de l'autre. Le plus efficace est donc d'activer UPDATE et ACCESS ; on visualise les dates d'intervention en écriture sur le fichier, ainsi que les dates de lecture.

DES VALEURS PAR DÉFAUT

SET

CP/M Plus

Donner un mot de passe par défaut.

Syntaxe

SET [DEFAULT=xxx]

où xxx est le mot de passe par défaut ; ce mot de passe sera utilisé automatiquement si un fichier est protégé par le même mot.

Ordre transitoire.

Exemple

```
A>SET TEXTE2.TXT [PASSWORD=TXT2 PROTECT=READ]

A:TEXTE2  .TXT  Protection = READ, Password = TXT2

A>TYPE TEXTE2.TXT


Password: TXT2
Fichier TEXTE2

A>SET [DEFAULT=TXT2]

Default password = TXT2


A>TYPE TEXTE2.TXT
Fichier TEXTE2

A>SET [DEFAULT=TST]

Default password = TST

A>TYPE TEXTE2.TXT


Password: TXT2
Fichier TEXTE2
A>
```

Le fichier TEXTE2.TXT est protégé par le mot de passe TXT2 dans le premier ordre. On voit dans l'ordre TYPE que le mot de passe est demandé.

On place (ordre SET [DEFAULT) le mot TXT2 comme mot de passe par défaut.

L'ordre de listage du fichier protégé par le même mot se fait sans demande du mot de passe.

Si le mot de passe par défaut est différent (ordre SET [DEFAULT = TST]), la machine redemande le mot de passe.

Le truc

Si l'on a l'habitude de travailler sur des fichiers protégés, on utilise en général le même mot ; il est plus pratique (et plus sûr) de donner ce mot en début de séquence. Dès que l'on éteindra l'ordinateur, tous les fichiers seront à nouveau protégés.

SETDEF

Nommer un lecteur de disquette par défaut.

Syntaxe

SETDEF n:

Ordre non résident.

Exemple

```
A>SETDEF B:

Drive Search Path:
1st Drive            - B:


A>DATE
Fri 10/18/85 12:18:54
A>dir
A: C10CPM3  EMS : ESS       BAK : ESS            : TEXTE3
A: SET      COM : TEXTE2    BAK : SETDEF    COM : DIR       COM
A: PIP      COM : ED        COM
A>dir b:
B: SET      COM : PIP       COM : DEBUT     SUB : SHOW      COM
B: SUBMIT   COM : C10CPM3   EMS : LIST      BAK : ED        COM
B: ERASE    COM : DIR       COM : LIST      SUB : TEXTE2
B: DATE     COM
A>
```

Le lecteur B est pris comme lecteur par défaut.

L'ordre DATE est exécuté immédiatement, alors que les ordres DIR vérifient que le fichier DATE.COM se trouve sur le lecteur B.

TOURS DE MAIN

LES ZONES USER

La disquette peut être séparée en zones USER ; dix zones portant le numéro 0 à 9 sont disponibles.

Passer d'une zone à l'autre

L'ordre USER suivi du numéro de la zone permet de rendre la zone active.
Exemple : USER 3 permet de passer en zone USER 3 sur le disque actif.
L'ordre xn : où x est le numéro d'une zone et n la lettre d'un lecteur permet de changer de lecteur et de se retrouver dans la zone x du nouveau lecteur.
Exemple : A> 2B: permet de passer du lecteur A au lecteur B zone 2.

Dans quelle zone est-on ?

Le prompt (signal de début de ligne) indique le lecteur actif et la zone USER active.
Exemple : A> lecteur A actif, zone 0 active ;
3A> lecteur A actif, zone 3 active.
La zone 0 est la zone active par défaut ; dans ce cas, le numéro de zone ne s'inscrit pas dans le prompt.

Pourquoi des zones ?

Le travail en zones permet à chaque utilisateur de travailler sur ses seuls fichiers ; il ne peut détruire ou endommager que ceux-là.
Le directory de sa propre zone est bien plus clair ; il comporte moins de fichiers, et uniquement les siens.

Comment travailler dans les zones ?

Une fois le changement fait vers sa zone, on travaille comme dans la zone 0, les sauvegardes et créations se faisant dans son propre espace de travail.

Des fichiers communs à toutes les zones

Les fichiers marqués de l'attribut SYS sont utilisables depuis toutes les zones USER ; ils ne sont modifiables et effaçables que de la zone dans laquelle ils résident.

Effacer des fichiers

On ne peut effacer des fichiers que dans la zone active. Le nettoyage d'une disquette est donc assez long ; il suppose de passer dans chaque zone.

Quelles sont les fichiers d'un disque, zone par zone ?

L'ordre SHOW [USERS] permet de connaître l'état d'encombrement des zones USER d'un disque.

L'ordre DIR [USER=ALL] permet d'obtenir la liste des fichiers d'une disquette, zone par zone.

Un truc

On a l'habitude de créer des zones USER sur les disques d'application ; le programme se trouve dans la zone 0, il est placé en attribut SYS. Les autres zones sont réservées aux utilisateurs qui travaillent tous sur le même programme, mais sur leurs propres données.

Exemple : en traitement de textes, le programme sera utilisé par tous, les textes frappés seront dans la zone de chaque rédacteur.

ENCHAINER DES PROCÉDURES – DÉMARRER AUTOMATIQUEMENT DES PROCÉDURES

Tout d'abord, CP/M Plus permet de donner plusieurs ordres à la fois, pourvu qu'ils soient séparés par un signe !

Par exemple :

```
DATE SET!DATE!DIR
```

enchaînera les trois ordres les uns derrière les autres.

CP/M Plus permet en outre d'automatiser des procédures de trois manières :

- les fichiers SUBMIT qui permettent de lancer des procédures en donnant des arguments, soit dans l'ordre lui-même (notion de variables), soit dans le fichier de commandes (notion de constantes) ;
- l'ordre GET permet de donner des ordres résidents dans un fichier ; les possibilités sont très voisines de celles des fichiers SUBMIT ;
- le fichier PROFILE.SUB est un fichier très particulier. Créé par l'utilisateur, il contient une suite d'ordres CP/M. Au démarrage, le système teste la présence de ce fichier ; s'il existe, il est exécuté automatiquement. Il se comporte un peu comme une "clé électronique" qui démarre le système et lance les ordres élémentaires.

Par exemple, on a vu que la date est un élément important, surtout si le directory est initialisé en datation (ordre INITDIR).

Il est donc important que réside dans la disquette un fichier PROFILE.SUB qui, créé à l'aide de l'éditeur ED ou de l'ordre PIP PROFILE.SUB=CON:, contiendra au

moins l'ordre DATE SET. De ce fait, toute mise sous tension obligera l'utilisateur à entrer la date et l'heure.

Dans les procédures de communications, que ce soit pour l'utilisation d'une imprimante série ou pour les communications par modem, il est important qu'un fichier PROFILE.SUB lance la configuration du port série: ordre DEVICE déterminant la vitesse de transmission et le protocole éventuel.

UTILISER LA FONCTION D'ARCHIVAGE

La fonction d'archivage permet de n'effectuer des sauvegardes que pour les fichiers qui ont été modifiés depuis la dernière sauvegarde.

Cette fonction est la combinaison de deux ordres :

- SET fich?* [ARCHIVE=ON] qui initialise le marqueur d'archive pour le ou les fichiers nommés ; le nom de fichier admettant des ambiguïtés, tous les fichiers d'un disque peuvent donc être marqués si fich vaut *.*;
- PIP n:=p:fich?*[A] l'option A pour archive ne sauvegardera sur le disque n que les fichiers correspondant au nom ou à l'ambiguïté donnée qui ont été modifiés depuis la dernière opération de sauvegarde.

Exemple

```
A>SET TEXTE*.*[ARCHIVE=ON]

A:TEXTE1  .    set to directory (DIR), Read Write (RW) A
A:TEXTE3  .    set to directory (DIR), Read Write (RW) A
A:TEXTE2  .    set to directory (DIR), Read Write (RW) A


A>PIP B:=TEXTE*.*[A]

ERROR: FILE NOT FOUND  - A:TEXTE???.???


A>ED TEXTE2
     : *i
    1:  On ajoute une ligne
     : *e

A>PIP B:=TEXTE*.*[A]

COPYING -
TEXTE2

A>
```

On suit les opérations de marquage (ordre SET) ; la première tentative de sauvegarde se solde par un échec, le système ne reconnaissant pas les fichiers puisqu'ils n'ont pas été modifiés.

On modifie avec l'éditeur ED le contenu du fichier TEXTE2 ; la tentative de sauvegarde archive ne sauvera que ce dernier fichier.

Le truc

Cette possibilité est à utiliser systématiquement avec les fichiers de données importants comme les fichiers d'adresses.

A PROPOS DE PROTECTIONS

La protection des fichiers offerte par CP/M Plus est très sophistiquée ; sa puissance suppose certaines précautions :

- Utilisez un nombre minimum de mots de passe, sinon vous risquez de les oublier et vous serez dans l'impossibilité d'accéder à vos fichiers.
- Si vos fichiers sont confidentiels, évitez de laisser la liste écrite de vos mots de passe à proximité ou même sur un fichier texte de la disquette.
- Lorsque vous frappez votre code, il ne s'affiche pas à l'écran ; il est cependant facile de lire ce que vous frappez sur le clavier, par-dessus votre épaule.
- Lorsque vous protégez vos fichiers avec le même code (au moins pour une gamme homogène d'applications), utilisez la fonction SET DEFAULT=mot de passe, ce qui vous fera gagner du temps et vous évitera d'avoir à refrapper fréquemment votre mot de passe.
- L'ordre de protection maximum d'un fichier :

  SET fich?* [RO,SYS,PASSWORD= mot de passe, PROTECT=READ]
- L'ordre de déprotection maximum :

  SET fich?* [RW,DIR,PROTECT=NONE]
- Utilisez un mot de passe correspondant à une mnémonique qui vous est personnelle, choisissez-le d'une longueur maximale, dans la limite de la longueur acceptée, soit 8 caractères.

L'ÉDITEUR ED.

L'éditeur de lignes ED est certes un peu archaïque par rapport aux éditeurs pleine page des machines actuelles ; il a cependant le mérite d'être fiable et d'être livré avec CP/M. Il est donc intéressant de savoir l'utiliser.

Deux modes de fonctionnement :

- **Le mode commande** est caractérisé par un astérisque (*) en début de ligne ;
- **Le mode insertion** est caractérisé par un numéro, seul, en début de ligne.

On passe du mode commande au mode insertion en frappant la lettre "i" ou la lettre "I" (dans ce dernier cas, seules des majuscules serons acceptées).

On passe du mode insertion au mode commande en frappant CTRL-Z.

Une seule ligne est active – c'est le principe de l'éditeur de lignes –. Son numéro figure en début de ligne. Si on est en insertion dans le texte, le numéro de la ligne active est celui marqué par un astérisque.

Le plus simple pour faire connaissance avec ED est encore de le manipuler et de faire ses propres expériences.

TABLEAU RÉCAPITULATIF/ÉDITEUR ED ORDRES PRINCIPAUX

☐ **nA**	(**A**ppend, Apporte) Place dans la mémoire de l'éditeur n lignes du fichier édité.
☐ **B**	(**B**egin) Place le pointeur de l'éditeur sur la première ligne de la mémoire d'édition.
☐ n**T**	Affiche n lignes de la mémoire d'édition, le pointeur restant en place (**T**exte). -n**T** affiche les lignes en remontant à partir du pointeur.
☐ **I**	Début d'**I**nsertion de texte au niveau du pointeur.
☐ **CTRL Z**	Fin d'insertion.
☐ n**L**	Avance de n **L**ignes dans la mémoire d'édition. -n**L** recule de n Lignes dans la mémoire d'édition à partir du pointeur.
☐ **F**texte	(**F**ind) Recherche le texte et place le pointeur immédiatement après.
☐ **S**texte1**CTRLZ**	Texte 2. Recherche texte 1 et lui substitue texte 2 (**S**ubstitue).
☐ n**K**	(**K**ill) Détruit n Lignes à partir du pointeur. -n**K** détruit n Lignes en remontant, à partir du curseur.
☐ n**C**	Déplacement de n **C**aractères. -n**C** déplace en reculant.
☐ n**D**	**D**étruit n caractères. -n**D** détruit en reculant.
☐ **Q**	**Q**uitter l'éditeur sans mise à jour.
☐ **E**	Quitter l'éditeur avec sauvegarde de la version mise à jour (**E**xit).
☐ **O**	Revient en mode édition au fichier **O**riginal en ignorant les modifications que l'on vient de faire.
☐ **0V**	Donne la place restant disponible en édition par rapport à la place initialement disponible dans la mémoire de l'éditeur.

Remarque. – Seules les commandes essentielles sont décrites ici. Ces commandes sont reçues par l'éditeur à la suite du message de disponibilité : *

INTRODUIRE UN PREMIER TEXTE DANS UN FICHIER ED

```
A>ED TEXTE

NEW FILE
      : *I
    1:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
    2:  PAR L'EDITEUR ED.
    3: ↑Z
      : *B
    1: *T
    1:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
    1: *#T
    1:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
    2:  PAR L'EDITEUR ED.
    1: *E

A>
```

La réponse NEW FILE signale que le fichier nommé n'existe pas encore et qu'il est donc créé.

- ☐ Les commandes **I** : indique une insertion de texte en majuscules.
  i : permet l'introduction minuscules et majuscules.
- ☐ L'ordre **CTRL Z** : indique la fin d'insertion.
- ☐ La commande **B** : place le pointeur de travail au début de la mémoire d'édition.
- ☐ La commande **T** : affiche une ligne de texte.
- ☐ La commande **#T** : affiche tout le contenu de la mémoire d'éditeur.
- ☐ La commande **E** : indique la fin de la séquence de travail avec sauvegarde du travail dans le fichier de nom TEXTE.

La fin de chaque ligne introduite est marquée par la touche ENVOI (RETURN, ENTER, LE...).

La commande E a pour effet de sauver le travail sur disque ; il est donc nécessaire de vérifier avant de la lancer, la fermeture du lecteur et la présence de la disquette.

Les touches control sous ED

Certaines touches contrôlées ont un effet particulier sous le programme éditeur ED :

- ☐ **CTRL C** : abandon de l'éditeur et retour sous CP/M.
- ☐ **CTRL E** : retour-chariot sans fin de ligne éditeur. La ligne éditeur peut comprendre plusieurs lignes physiques d'écran ou d'imprimante.
- ☐ **CTRL H** : détruit le dernier caractère frappé (équivaut à la touche RUB de certaines machines).
- ☐ **CTRL U** : détruit la ligne entière en cours de frappe.

INSERTION DE LIGNES DANS UN TEXTE

```
A>ED TEXTE
       : *#AB#T
     1:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
     2:  PAR L'EDITEUR ED.
     1: *I        ECRITURE D'UN TEXTE AVEC ED
*B#T
     1:           ECRITURE D'UN TEXTE AVEC ED
     2:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
     3:  PAR L'EDITEUR ED.
     1: *
     2:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
     2: *I        ---------------------------
*B#T
     1:           ECRITURE D'UN TEXTE AVEC ED
     2:           ---------------------------
     3:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
     4:  PAR L'EDITEUR ED.
     1: *E

A>DIR
A: TEXTE    BAK : TEXTE         : ED       COM
A>
```

La commande #A charge dans la mémoire d'édition toutes les lignes du fichier disque TEXTE (la commande 2A aurait chargé les deux premières lignes).

La commande B place le pointeur au début de la mémoire.

La commande #T liste à l'écran la totalité de la mémoire (la commande 2T aurait listé deux lignes à partir du pointeur ; la commande -2T aurait listé deux lignes à partir du pointeur, en remontant ; ici, le pointeur étant au début, cette commande n'aurait eu aucun effet).

La commande I permet d'insérer une ligne, l'envoi (R) indique la fin d'insertion.

B#T permet de vérifier l'insertion.

Une seconde ligne est insérée, le listage vérifie l'existence de quatre lignes de texte.

E termine l'édition et sauve sous le nom TEXTE le fichier modifié.

L'ordre CP/M DIR permet de voir que l'ancien fichier est conservé sous l'extension .BAK (BACKUP).

INSERTION D'UN TEXTE AU DÉBUT D'UNE LIGNE

```
A>ED TEXTE
      : *#AB#T
    1:            ECRITURE D'UN TEXTE AVEC ED
    2:            ---------------------------
    3:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
    4:  PAR L'EDITEUR ED.
    1: *3L
    4: *IPREFERE ↑Z

*B#T


    1:            ECRITURE D'UN TEXTE AVEC ED
    2:            ---------------------------
    3:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
    4:  PREFERE PAR L'EDITEUR ED.
    1: *E

A>
```

La séquence #AB#T charge le fichier édité dans la mémoire, place le pointeur au début et liste la totalité du texte.

Les ordres d'édition peuvent être groupés :

- La commande 3L place le pointeur trois lignes plus loin.
- La commande IPREFERE CTRLZ insère le texte PREFERE à partir de la position du curseur (I=début d'insertion, CTRLZ=fin).
- La commande B#T permet de lister toute la mémoire depuis le début.

E termine la séquence en sauvant sur disque la nouvelle version.

AJOUTER DU TEXTE A LA FIN

```
A>ED TEXTE
      : *#AB#T
     1:          ECRITURE D'UN TEXTE AVEC ED
     2:          ---------------------------
     3:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
     4:  PREFERE PAR L'EDITEUR ED.
     1: *-BI
     5:  L'EDITEUR PERMET DE SUPPRIMER FACILEMENT UNE LIGNE
     6:  DE TEXTE
     7: ↑Z
      : *-2L
     5: *K
     5: *B#T
     1:          ECRITURE D'UN TEXTE AVEC ED
     2:          ---------------------------
     3:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
     4:  PREFERE PAR L'EDITEUR ED.
     5:  DE TEXTE
     1: *4LK
      : *Q

Q-(Y/N)?Y

A>
```

L'ordre -BI cumule le passage du pointeur en fin de mémoire (-B) et l'ordre d'insertion début (I). A la fin du texte inséré, on termine par CTRL Z.

L'ordre -2L fait remonter le curseur de deux lignes (7-2=5).

L'ordre K détruit la ligne ce qui est vérifié par la séquence B#T qui liste toute la mémoire.

La séquence 4LK fait avancer le pointeur de quatre lignes (1+4=5) et détruit la ligne (donc la 5).

L'ordre O permet de quitter l'édition sans sauver la version de travail (ici, le texte de fin était le même que celui de début, donc inutile de le sauver).

```
A>TYPE TEXTE
         ECRITURE D'UN TEXTE AVEC ED
         ---------------------------
CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
PREFERE PAR L'EDITEUR ED.

A>
```

Sous CP/M l'ordre TYPE permet de lister en clair un fichier créé sous éditeur ED.

INSÉRER DU TEXTE A L'INTÉRIEUR D'UNE LIGNE

```
A>ED TEXTE
      : *#AB#T
     1:          ECRITURE D'UN TEXTE AVEC ED
     2:          ---------------------------
     3:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
     4:  PREFERE PAR L'EDITEUR ED.
     1: *3L
     4: *FEDITEUR
     4: *I DE TEXTE↑Z
*B#T
     1:          ECRITURE D'UN TEXTE AVEC ED
     2:          ---------------------------
     3:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
     4:  PREFERE PAR L'EDITEUR DE TEXTE ED.
     1: *E
```

Passons une dernière fois sur la séquence #AB#T qui est une séquence systématique au début de tout travail sous ED.

L'ordre 3L place le pointeur trois lignes plus loin donc ligne 4.

L'ordre FEDITEUR (pour Find EDITEUR CaD Cherche le mot EDITEUR) place le pointeur immédiatement *après* le mot. Sans l'ordre 3L, le pointeur se serait placé au même endroit puisque le mot ne se trouve pas dans les lignes précédentes.

L'ordre I DE TEXTECTRLZ permet d'insérer le texte qui est entre les marques de début et de fin d'insertion (I et CTRLZ) donc les mots DE TEXTE.

L'ordre B#T permet de vérifier à l'écran le contenu de la mémoire de travail et donc si l'insertion a bien été faite.

REMPLACER UN TEXTE PAR UN AUTRE

```
A>ED TEXTE
      : *#AB#T
     1:          ECRITURE D'UN TEXTE AVEC ED
     2:          ---------------------------
     3:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
     4:  PREFERE PAR L'EDITEUR DE TEXTE ED.
     1: *SL'EDITEUR↑ZLE PROGRAMME D'EDITION
     4: *B#T
     1:          ECRITURE D'UN TEXTE AVEC ED
     2:          ---------------------------
     3:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
     4:  PREFERE PAR LE PROGRAMME D'EDITION DE TEXTE ED.
     1: *E
```

L'ordre SL'EDITEURCTRLZLE PROGRAMME D'EDITION signifie échanger (Substitute) à la suite de mots "L'ÉDITEUR" la suite de mots "LE PROGRAMME D'EDITION".

L'ordre B#T permet de lister toute la mémoire depuis le début et donc de vérifier que la substitution s'est correctement effectuée.

On remarque qu'une recherche ou une substitution peut se faire quelle que soit la position du pointeur pourvu qu'il soit placé en avant du mot à rechercher ou à substituer.

La substitution faite, le pointeur se trouve sur la ligne et à la fin de la chaîne substituée.

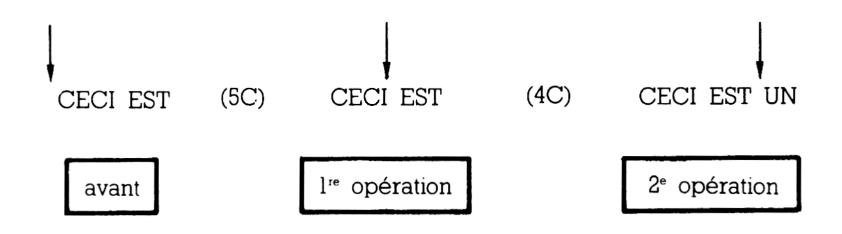
L'ordre E sauve la version modifiée du texte.

DÉPLACER LE CURSEUR SUR UNE LIGNE – DÉTRUIRE DES CARACTÈRES

```
A>ED TEXTE
      : *#AB#T
    1:            ECRITURE D'UN TEXTE AVEC ED
    2:            -------------------------
    3:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
    4:  PREFERE PAR LE PROGRAMME D'EDITION DE TEXTE ED.
    1: *2L
    3: *5C
    3: *T
EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
    3: *4C
    3: *T
UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
    3: *9D
    3: *B#T
    1:            ECRITURE D'UN TEXTE AVEC ED
    2:            -------------------------
    3:  CECI EST UN TEXTE INTRODUIT DANS VOTRE ORDINATEUR
    4:  PREFERE PAR LE PROGRAMME D'EDITION DE TEXTE ED.
    1: *E
```

L'ordre 2L pousse le pointeur de 2 lignes.

L'ordre 5C pousse le pointeur de 5 caractères (il était en début de ligne : il passe donc au 5ᵉ caractère, l'espace entre CECI et EST :



L'ordre T liste la mémoire à partir de la position du pointeur.
L'ordre 9D détruit 9 caractères à partir du pointeur.
La séquence B#T permet de lister la totalité de la mémoire modifiée.
E sauve sur disque la version modifiée.

PRATIQUE DE L'ÉDITEUR

Courrier automatique

Ce petit tour de main vous permet de créer une lettre personnalisée à partir d'une lettre standard.

Le procédé fait appel aux possibilités de l'éditeur ED livré en série avec le système d'exploitation CP/M.

Nous utiliserons les possibilités de création d'un texte, puis éventuellement celles de correction mais surtout la substitution d'une chaîne par une autre que permet l'éditeur grâce à sa commande S.

Le principe sera donc de créer une lettre dont les valeurs susceptibles de changement seront précisées à leur place dans le texte, la personnalisation du courrier n'étant que la substitution de ces chaînes de caractères par celles souhaitées.

Nous utiliserons pour finir la particularité de ED de conserver la version originale du texte sur disque pour la renommer après le travail et de ne conserver que la version de base réutilisable pour un autre courrier.

Créer le texte de la lettre

```
A>ED B:LETTRE

NEW FILE
     : *;
 1:
 2:
 3:
 4:
 5:                                                Paris le 10/10/85
 6:
 7: Société Dupont
 8: 17, rue des Roses
 9: 75025 Paris
10:
11:                                                à l'attention de
12:                                                NOM1
13:                                                ADR1
14:                                                COVI1
15:
16:             Monsieur,
17:
18:      Nous avons bien reçu votre récent courrier, et nous
19:accusons réception de votre commande.
20:
21:      Afin de réduire votre attente, nous vous serions
22:reconnaissant de bien vouloir nous adresser la somme
23:de SOM1, correspondant aux frais de port et au montant de
24:votre commande.
25:
26:      Dans cette attente, nous vous prions de croire, NOM1,
27:en nos sincères salutations.
28:
29:
30:
31:                                               Le service commandes
32:↑Z
33:*E
```

Remarque. – Le signe d'insertion est un i minuscule, ce qui permet d'écrire en minuscules ou majuscules. Si on met le I majuscule, toutes les minuscules frappées seront changées en majuscules.

La ligne 32 est le control Z de fin de fichier.

L'ordre E de la ligne 33 sauve le fichier et clôt l'introduction.

Modifier automatiquement les paramètres

```
A>DIR B:
B: LETTRE
A>ED B:LETTRE
       : *#A
      1: *SDAT1↑Z10/10/82
      5: *#SNOM1↑ZMONSIEUR DUMUR

BREAK "#" AT R
    27: *B
     1: *SADR1↑Z25, RUE BLEUE
    12: *SCOVI1↑Z17216  LA PLAINE
    13: *SSOM1↑Z165.50 F
    23: *E

A>
```

L'ordre #A charge la lettre en mémoire.

La ligne 1 remplace DAT1 par 10/10/82 (notez la place de Control Z qui finit la première chaîne).

En ligne 5, le signe # avant SNOM1 généralise la substitution qui suit à toutes les chaînes NOM1 rencontrées ; de ce fait, une fois la 3[e] substitution faite, la machine indique qu'elle est arrêtée en ligne 27.

L'ordre B remonte le pointeur au début du fichier pour la suite des substitutions.

La ligne 23 termine cette séquence.

On se trouve maintenant avec une lettre personnalisée qui est rangée dans le secteur B sous le nom de "LETTRE".

L'éditeur a conservé la version précédente (non personnalisée) sous le nom "LETTRE.BAK".

Les numéros de ligne sont les témoins de la substitution, par exemple, la substitution de la ligne 1 place en début de seconde ligne le nombre 5 indiquant que la dernière substitution s'est faite ligne 5.

Taper la lettre

```
TYPE B: LETTRE

                                        paris le 10/10/85


Société Dupont
17, rue des Roses
75025 Paris

                                        à l'attention de
                                        Monsieur Dumur
                                        25, rue Bleue
                                        17216 La Plaine


         Monsieur,
     Nous avons bien reçu votre récent courrier, et nous
accusons réception de votre commande.

     Afin de réduire votre attente, nous vous serions
reconnaissant de bien vouloir nous adresser la somme de 165,50
F, correspondant aux frais de port et au montant de votre
commande.

     Dans cette attente, nous vous prions de croire, Monsieur
Dumur, en nos sincères salutations.



                                         Le service commandes
```

Notre ordinateur permettant la double sortie (vers l'imprimante en même temps que l'écran), l'ordre TYPE suffit, sinon il faut auparavant taper l'ordre Control P qui fera le même effet.

Pour finir

```
A>
A>ERA B:LETTRE
A>REN B:LETTRE=B:LETTRE.BAK
A>
```

Cette séquence permet de détruire la lettre personnalisée et de renommer la version standard (nommée avec l'extension .BAK par ED) du nom d'origine (LETTRE).

Ceci permet donc une autre utilisation pour une nouvelle lettre.

```
A>ED B:LETTRE
       : *I
     1 : ↑Z
       : *B
```

Vous avez remarqué que tous les mots remplacés sont en majuscules, ED se met en effet en insertion majuscules si rien ne lui a été précisé.

La séquence proposée ici, placée après la ligne #A met donc en insertion minuscules ou majuscules.

Annexe 1
Fonctions des ordres CP/M correspondant à la programmation avancée

Ces ordres sont en fait des outils de programmation destinés aux développeurs. pratiquant la programmation en assembleur Z80. Ils ne sont donc pas indispensables à la majorité des utilisateurs. Nous ne les développerons pas, invitant le lecteur à consulter des ouvrages spécialisés dans la programmation du Z80.

- ☐ **DUMP** Affiche le contenu d'un fichier en hexadécimal, en précisant les adresses de chaque octet et en donnant en fin de ligne la traduction ASCII de ces octets.
- ☐ **HEXCOM** Génère un fichier de commande (de type .COM), à partir d'un fichier hexa (de type .HEX) créé par un autre utilitaire.
- ☐ **LINK** Utilise des modules objets créés par un macro-assembleur, pour en faire des fichiers commandes en utilisant des fichiers bibliothèques. De nombreuses options à l'ordre en font un outil indispensable dans la chaîne de création d'un fichier commande.
- ☐ **LIB** Sert à la gestion des librairies de programmes sources.
- ☐ **MAC** Macro-assembleur utilisant un fichier .ASM en entrée, et générant trois fichiers (.HEX, .PRN et .SYM) de même nom, en sortie.
- ☐ **PATCH** Permet de modifier le programme système CP/M.
- ☐ **RMAC** Macro-assembleur translatable, créant un fichier .REL en place du fichier .HEX de MAC permettant d'éditer des liens.
- ☐ **SID** Utilitaire de mise au point des programmes, proche de DDT de CP/M2, il permet d'intervenir directement, au niveau des octets d'un programme, sur le fichier disquette.
- ☐ **XREF** Permet de lister les références croisées des variables d'un programme issu de MAC ou de RMAC.

Annexe 2
Tableaux récapitulatifs des ordres CP/M Plus

Les ordres entre parenthèses sont des ordres transitoires ; le fichier de type COM correspondant doit exister sur l'unité active pour qu'ils soient opérationnels.

Par exemple, l'ordre DIR [FULL] ne sera fonctionnel que si le fichier DIR.COM se trouve sur l'unité active.

Certains ordres disponibles sous CP/M Plus ne figurent pas dans cette liste ; il s'agit d'ordres très particuliers qui feraient l'objet à eux seuls d'un ouvrage (le BASIC GSX par exemple). En règle générale, nous avons retenu les ordres qui font l'utilisation courante de CP/M Plus et qui couvrent en moyenne plus de 95 % des besoins.

Les abréviations suivantes sont utilisées :

☐ **fich** : Nom de fichier sans caractère ambigu dans son nom.

☐ **fich*?** : Nom de fichier acceptant les caractères ambigus.

☐ **log:** : Nom d'unité logique ;

CONOUT: Sortie console (le plus souvent un écran) ;
CONIN: Entrée console, (le plus souvent un clavier) ;
AUXIN: Entrée auxiliaire ;
AUXOUT : Sortie auxiliaire ;
LST : Imprimante.
Nom d'unité physique (en général) ;

☐ **phys** : IO Interface série ;
CRT Écran/clavier ;
LPT Imprimante ;
NULL Annule la sortie physique.

☐ **physpip** : Nom d'unité physique (en général) ;
CON: L'écran/clavier ;
AUX: Sortie auxiliaire ;
LST: Imprimante ;
PRN: Simulation d'imprimante télétype (avec numéros en début de ligne).

(PIP)

- Débute une séquence de transferts sur questionnement.
- A:=B:fich?* — Copie tous les fichiers du disque B, compte tenu des caractères ambigus, sur le disque.
- A:fich1=B:fich2 — Copie sur le disque A, et sous le nom fich1, le fichier de nom fich2 se trouvant sur le disque.
- A:FICH1=B:FICH2,fich3,_etc. — Copie sous le nom fich1 et sur le disque A le fichier fich2, suivi du fichier fich3. Le fichier dont le nom est à droite du signe = est donc la juxtaposition dans l'ordre des fichiers dont les noms figurent à droite.
- physpip:=fich — Copie sur l'unité physique physpip (CON:, LST:, AUX:, PRN:) le contenu du fichier fich.
- fich=physpip: — Tous les caractères issus de l'unité physique (en général CON: ou AUX:) sont stockés dans le fichier fich jusqu'à l'émission du caractère de fin de fichier (^Z).
- physpip1:=physpip2: — Tous les caractères émis par l'unité physique physpip2 sont transmis à l'unité physique physpip2.

Options :

- [A] — Agit sur les seuls fichiers modifiés depuis la dernière copie.
- [C] — Demande d'accord avant chaque copie (Confirm).
- (Dx] — Détruit tous les caractères placés après la xième colonne.
- [E] — Affiche le double du transfert à l'écran (Echo).
- [F] — Ne copie pas les caractères de fin de page.

[Gx] — La disquette source (ou la disquette destination selon la place de l'option) travaille sur la zone USER numéro x.

[H] — Copie sous forme hexadécimale, en vérifiant caractère par caractère la conformité avec la norme Intel (Hexa).

[I] — Copie en hexadécimal en ignorant le caractère hexa 00.

[K] — La copie se fait sans affichage du nom du fichier en cours de copie (Kill).

[L] — Tous les caractères majuscules du fichier source : sont transformés en majuscules au cours de la : copie.

[N] — Chaque début de ligne est précédé, sur la copie, du numéro de cette ligne. L'option N2 donne une numérotation à six chiffres ; des zéros remplacent les chiffres non significatifs (numéros).

[O] — Copie de fichiers en codes machines, même les caractères ^Z sont copiés sans stopper la copie (fichiers Objets).

[Px] — Fixe la longueur des pages en sortie à x lignes ; par défaut, x vaut 60. Toutes les x lignes, PIP envoie un saut de page (Form Feed) sur l'unité.

[Qch^Z] — L'option Q permet d'arrêter la copie après que PIP ait trouvé la chaîne de caractère ch. La chaîne qui figure dans l'ordre doit se terminer par le caractère ^Z (Quit).

[R]——Permet de copier les fichiers qui possèdent l'attribut SYS.

[Sch^Z]——Commence la copie dès la chaîne ch ; la chaîne mentionnée dans l'ordre se termine par le caractère ^Z (option souvent associée à l'option Q pour transférer une partie d'un fichier).

[Tx]——La copie se fait en transformant la longueur des caractères de tabulation (^I) rencontrés en x espaces.

[U]——Transforme les minuscules rencontrées en majuscules.

[V]——Dans le cas de copie de disquette à disquette, PIP vérifie que la copie est bien conforme à l'original.

[W]——Permet, dans le cas où le nom du fichier destination existe déjà sur la disquette, de le détruire, même s'il est protégé (R/O).

[Z]——Copie sans utiliser le bit de parité pour contrôle.

Commande	Option	Effet
(PUT)	CONSOLE FILE fich	Le fichier fich se remplit de toutes les informations envoyées à l'écran par l'ordre suivant.
	PRINTER FILE fich	Le fichier fich se remplit de toutes les informations envoyées à l'imprimante par l'ordre suivant.
	PRINTER PRINTER	Les sorties vers l'imprimante sont à nouveau dirigées à l'imprimante.
	CONSOLE CONSOLE	Les sorties écran sont à nouveau dirigées vers l'écran.
RENAME	fich1=fich2	Le fichier de nom fich2 prend le nom fich1.
(RENAME)		Le système demande le nom du fichier à renommer, le nouveau nom à lui donner, puis effectue l'opération.
	fich?*1=fich?*2	Tous les fichiers dont le premier nom ambigu existe sont renommés dans le second nom ambigu.

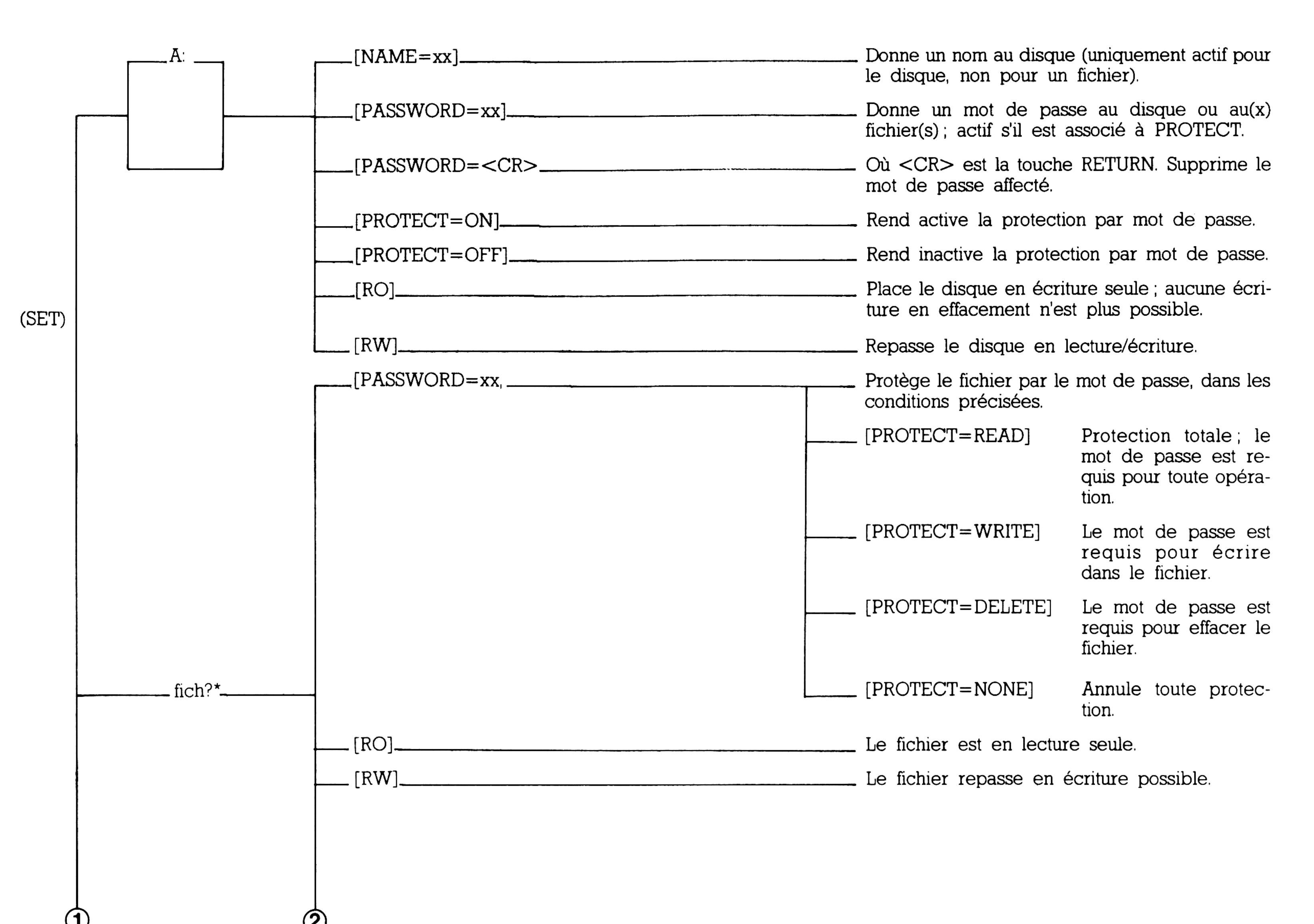
Commande	Option	Description
(DATE)		Donne la date et l'heure.
	C	Donne date et heure en permanence.
	MM/JJ/AAHH:MM:SS	Met à jour la date et l'heure.
	SET	Met à jour date et heure sur questionnement.

Commande	Option	Description
(DEVICE)	NAMES	Donne la liste des dispositifs physiques reconnus de CP/M.
	VALUES	Donne la liste des dispositifs physiques et les dispositifs logiques (connus de CP/M) qui les commandent.
		Cumule les options NAMES et VALUES et propose des modifications.
	CRT	Visualisation de l'état de l'unité physique clavier/écran.
	CON	Visualisation de l'état de l'unité logique clavier et de l'unité logique écran.
	log:=phys[OPTION]	Affecte l'unité physique à l'unité logique avec (ou sans) option.
	CONSOLE [PAGE]	Donne les caractéristiques de l'affichage écran (lignes × colonnes).
	CONSOLE [COLUMNS=XX−LINES=YY]	Modifie les caractéristiques de l'affichage (met à XX colonnes et à YY lignes).

Commande	Option	Description
DIR		Donne la liste simple de tous les fichiers du disque actif.
	B:	Même action mais pour les fichiers du disque B.
		Liste des fichiers d'extension .COM.
		Liste des fichiers dont le nom commence par la lettre A.

Commande	Option	Description
(DIR)	[ATT]	Donne les attributs des fichiers.
	[DATE]	Donne les attributs date et heure des fichiers si le directory a été modifié et si l'ordre SET correspondant a été exécuté.
	[DIR]	Liste les fichiers d'attribut DIR.
	[DRIVE=ALL]	Liste les fichiers de tous les lecteurs connus du système.
	[DRIVE=A,B,.]	Liste les fichiers de tous les lecteurs nommés.
	[EXCLUDE]	Liste les fichiers qui ne répondent pas à la spécification précisée.
	[FULL]	Liste les fichiers en donnant tous les renseignements (noms classés par ordre alphabétique, attributs, taille...).
	[NOSORT]	Liste les fichiers dans l'ordre du directory (non reclassés par ordre alphabétique).
	[RO] ou [RW]	Liste les fichiers mis en lecture seule ou ceux qui sont en lecture/écriture.
	[SIZE]	Liste avec la taille en kilo-octets.
	[SYS]	Liste les fichiers ayant l'attribut SYS.
	[USER]=ALL]	Liste les fichiers de toutes les zones utilisateurs.
	[USER]=(0,1)	Liste les fichiers des zones nommées.

Commande	Paramètres	Description
DIRSYS	B: / fich?*	Donne la liste des fichiers affectés de l'attribut SYS.
(DUMP)	fich	Liste en hexadécimal avec son interprétation ASCII le fichier de nom fich ; fich ne peut pas contenir de caractères ambigus.
ERASE	fich*?	Détruit le fichier de nom fich ; fich peut contenir des ambiguïtés.
(ERASE)	fich*?	Détruit le fichier nommé. Si le nom de fichier contient un caractère ambigu, il y aura demande de confirmation (Y ou N).
	[C]	Détruit les fichiers un par un après confirmation (Y ou N).
(GET)	FILE fich	Donne en entrées le contenu du fichier fich pour le programme qui sera lancé.
	FILE fich [ECHO]	Il y a écho des entrées à l'écran.
	FILE fich [NO ECHO]	Les entrées ne sont pas répétées à l'écran.
	FILE fich [SYSTEM]	Le fichier est exécuté directement par le système comme une suite d'ordres.
	CONSOLE	Rend les entrées au clavier.
(HELP)		Entre en mode aide et propose la liste des mots expliqués.
	mot clé	Donne les explications sur le mot clé spécifié.
(INITDIR)	n:	Initialise le disque n en vue du marquage heure et date des fichiers.


(SET)
A:
[NAME=xx]
Donne un nom au disque (uniquement actif pour le disque, non pour un fichier).
[PASSWORD=xx]
Donne un mot de passe au disque ou au(x) fichier(s) ; actif s'il est associé à PROTECT.
[PASSWORD=<CR>
Où <CR> est la touche RETURN. Supprime le mot de passe affecté.
[PROTECT=ON]
Rend active la protection par mot de passe.
[PROTECT=OFF]
Rend inactive la protection par mot de passe.
[RO]
Place le disque en écriture seule ; aucune écriture en effacement n'est plus possible.
[RW]
Repasse le disque en lecture/écriture.
fich?*
[PASSWORD=xx,
Protège le fichier par le mot de passe, dans les conditions précisées.
[PROTECT=READ]
Protection totale ; le mot de passe est requis pour toute opération.
[PROTECT=WRITE]
Le mot de passe est requis pour écrire dans le fichier.
[PROTECT=DELETE]
Le mot de passe est requis pour effacer le fichier.
[PROTECT=NONE]
Annule toute protection.
[RO]
Le fichier est en lecture seule.
[RW]
Le fichier repasse en écriture possible.
1
2

Option	Description
[SYS]	Le fichier n'apparaît pas dans le directory simple.
[DIR]	Le fichier apparaît au directory.
[ARCHIVE=ON]	Rend active la fonction archive de PIP [A] ; ces fichiers ne seront recopiés par PIP que s'ils ont subi un changement.
[ARCHIVE=OFF]	Rend la fonction archive inactive.
[CREATE=ON]	Active le dateur de fichiers pour toute création de fichiers (OFF le désactive).
[ACCESS=ON]	Active le dateur de fichiers pour toute lecture du fichier (OFF le désactive).
[UPDATE=ON]	Active le dateur de fichiers pour toute modification du fichier. (OFF le désactive).
[DEFAULT=xx]	Donne un mot de passe global par défaut pour tous les fichiers protégés du disque.

(SETDEF)

Option	Description
n:	Le système recherchera d'abord les fichiers sur l'unité n, qui devient l'unité par défaut. Il peut y avoir plusieurs unités par défaut.
[TEMPORARY=p:]	Les fichiers temporaires seront pris sur l'unité p ; l'unité par défaut ne change pas.
[ORDER=ext,ext1]	Ordre de priorité des extensions de fichiers lorsque plusieurs fichiers ont le même nom.
[DISPLAY]	Le système affiche le nom des programmes exécutés (NODISPLAY l'annule).
[PAGE]	L'affichage se fait par écran, lorsqu'il est trop long. NOPAGE donne un affichage en continu.

Syntaxe	Description
(SHOW) A:	Donne la place disponible sur le disque spécifié (ou actif, par défaut.).
(SHOW) A: [LABEL]	Donne le nom de la disquette s'il a été spécifié.
(SHOW) A: [USERS]	Donne les indications relatives aux zones USER de la disquette active.
(SHOW) A: [DIR]	Donne le nombre d'entrées directory restant disponibles.
(SHOW) A: [DRIVE]	Donne tous les renseignements relatifs à la disquette contenue sur le lecteur nommé.
STAT␣ (R)	Donne la place disponible sur le disque actif.
STAT␣ N: (R)	Donne la place disponible sur le disque N:
STAT␣ N: R/O (R)	Place provisoirement le disque N: en lecture seule.
STAT␣ FICHIER.EXT␣ $ R/O (R)	Place le fichier nommé en lecture seulement.
STAT␣ FICHIER.EXT␣ $ R/W (R)	Place le fichier nommé en lecture/écriture.
STAT␣ FICHIER.EXT␣ $ DIR (R)	Le fichier nommé est visible à l'ordre DIR.
STAT␣ FICHIER.EXT␣ $ SYS (R)	Le fichier nommé n'apparaît pas à l'ordre DIR.
STAT␣ FICHIER.EXT␣ (R)	Donne l'encombrement du fichier nommé.
STAT␣ DEV: (R)	Donne les caractéristiques des quatre périphériques gérés par CP/M.
STAT␣ VAL: (R)	Donne la syntaxe et les attributs possibles associés à l'ordre STAT.
STAT␣ USR: (R)	Donne le numéro de la zone utilisateur en service et celles qui ont été utilisées sur le disque.

(1)

Commande	Paramètre	Option	Description
	N:DSK: (R)		Donne toutes les caractéristiques physiques du disque N:
	CON:=CRT: (R)		Affecte à un des périphériques (ici CON) une caractéristique particulière (ici CRT).
(SUBMIT)	Fich		Exécute les ordres CPM contenus dans le fichier fich, les uns après les autres.
	fich arg1 arg2		Exécute les ordres du fichier fich en y intégrant les arguments arg1 arg2...
TYPE	fich		Affiche à l'écran le contenu texte du fichier fich.
(TYPE)	fich	[PAGE]	Affiche le fichier fich page par page s'il comprend plus de vingt-quatre lignes.
		[NOPAGE]	Affiche fich en continu.
USER	n		Passe dans la zone utilisateur n.

Annexe 3
Index des questions que vous vous posez

Opérations d'initialisation

Opérations sur les périphériques

Opérations sur la liste des fichiers (directory)

Opérations de base sur les fichiers

Opérations de transfert entre unités périphériques

Les options

Des informations sur le disque

Utiliser les attributs des fichiers

Donner des valeurs par défaut

Annexe 4
Index

CONSEILS DE LECTURE

Pour approfondir vos connaissances en BASIC 7.0 et mieux connaître le système du C 128, P.S.I. vous propose une palette d'ouvrages utiles.

POUR MAITRISER LE BASIC DU C 128

A paraître :

☐ **Commodore 128 : Méthodes pratiques** – Jacques Boisgontier (Éditions du P.S.I.)

Pour ceux qui ont déjà pratiqué un BASIC, voici un ouvrage de perfectionnement au BASIC 7.0. Un chapitre sur le CP/M Plus donne les principales commandes systèmes.

☐ **102 programmes pour Commodore 128** – Jacques Deconchat (Éditions du P.S.I.)

Les bases de la programmation en cinq niveaux de difficulté pour apprendre le BASIC 7.0 en jouant.

POUR MIEUX CONNAITRE LE SYSTÈME DU COMMODORE 128

Déjà paru :

☐ **Clefs pour Commodore 128 : 1-Mode 64** – Daniel-Jean David (Éditions du P.S.I.)

Mémento présentant synthétiquement le jeu d'instructions du 6510, les points d'entrée des routines système, les connecteurs, les messages d'erreurs, etc. Le livre de chevet du programmeur sur Commodore.

A paraître :

☐ **Clefs pour Commodore 128 : 2-Mode 128** – Jean-François Sehan (Éditions du P.S.I.)

Ce deuxième tome du *Clefs pour C 128* donne les adresses BASIC 7.0, l'architecture du CP/M Plus et toutes les informations indispensables au programmeur sur Commodore 128.

☐ **Désassemblage des ROM du Commodore 128** – Milton Bathurst (Datacap – Diffusé par les éditions du P.S.I.)

La ROM désassemblée et commentée pour programmeur chevronné.

CONSEILS DE L'AUTEUR POUR PROGRAMMER SOUS CP/M PLUS

☐ **CP/M Plus Operating System Programmer's Guide** (Digital Research)

☐ **Programmer's utilities guide for the CP/M Plus family of operating system** (Digital Research)

Achevé d'imprimer en janvier 1986
sur les presses de l'imprimerie Laballery et Cie
58500 Clamecy
Dépôt légal : janvier 1986
N° d'impression : 601045
N° d'édition : 86595-327-1
ISBN : 2-86595-327-0

Votre avis nous intéresse

- Pour nous permettre de faire de meilleurs livres, adressez-nous vos critiques sur le présent livre.
- Si vous souhaitez des éclaircissements techniques, écrivez-nous, nous adresserons votre demande à l'auteur qui ne manquera pas de vous répondre directement.

- Ce livre vous donne-t-il toute satisfaction?

___

___

___

___

___

- Y a-t-il un aspect du problème que vous auriez aimé voir abordé?

___

___

___

___

___

Comment avez-vous eu connaissance de ce livre?

☐ publicité
☐ catalogue
☐ boutique micro
☐ autres
☐ cadeau
☐ librairie
☐ exposition

Avez-vous déjà acquis des livres PSI?

lesquels? ___

___

qu'en pensez-vous? ___

___

___

Nom ___ Prénom ___ Age ___
Adresse ___
Profession ___
Centre d'intérêt ___

CATALOGUE GRATUIT

Vous pouvez obtenir un catalogue complet des ouvrages PSI, sur simple demande, ou en retournant cette page remplie à votre libraire, à votre boutique micro ou aux

Editions du PSI
BP 86
77402 Lagny-sur-Marne Cedex



SYSTEMES

CP/M PLUS SUR AMSTRAD CPC, PCW ET COMMODORE 128

CP/M Plus est un système d'exploitation conçu par Digital Research pour piloter des ordinateurs fonctionnant avec les microprocesseurs Z80 ou similaires. Il n'existe donc pas de différence fondamentale dans l'utilisation et la syntaxe des ordres de CP/M selon que l'on utilise une machine Amstrad CPC, PCW ou Commodore 128.

Cet ouvrage vous apprendra à vous servir au mieux de l'éditeur, à copier, à protéger ou à lister un fichier, à enchaîner plusieurs commandes CP/M, à formater ou à dupliquer un disque, etc.

Vous saurez également mettre à jour la date, l'heure de votre micro, modifier les caractéristiques de l'affichage écran, lister les fichiers protégés ou non...

Après lecture de cet ouvrage, CP/M Plus n'aura plus de secret pour vous.

PRIX : 100 FF

ISBN : 2-86595-469-2

9 782865 954698

ÉDITIONS P.S.I.
DIFFUSÉ PAR P.C.V. DIFFUSION
BP 86 - 77402 LAGNY-S/MARNE CEDEX – FRANCE

P.S.I.

CP/M PLUS SUR AMSTRAD CPC, PCW ET COMMODORE 128



Document numérisé avec amour par

AMSTRAD CPC

MÉMOIRE ÉCRITE



https://acpc.me/